Early Stage Integrated Parametric Ship Design

M. Bole, C. Forrest
Graphics Research Corporation Ltd, UK.

Abstract

Innovative ship design projects often require an extensive concept design phase to allow a wide range of potential solutions to be investigated, identifying which best suits the requirements. In these situations, the majority of ship design tools do not provide the best solution, limiting quick reconfiguration by focusing on detailed definition only. Parametric design, including generation of the hull surface, can model topology as well as geometry offering advantages often not exploited.

Paramarine is an integrated ship design environment that is based on an object-orientated framework which allows the parametric connection of all aspects of both the product model and analysis together. Design configuration is managed to ensure that relationships within the model are topologically correct and kept up to date. While this offers great flexibility, concept investigation is streamlined by the Early Stage Design module, based on the (University College London) Functional Building Block methodology, collating design requirements, product model definition and analysis together to establish the form, function and layout of the design.

By bringing this information together, the complete design requirements for the hull surface itself are established and provide the opportunity for parametric hull form generation techniques to have a fully integrated role in the concept design process. This paper explores several different hull form generation techniques which have been combined with the Early Stage Design module to demonstrate the capability of this design partnership.

Introduction

It is widely known that the decisions made in the earliest stages of design can have the greatest effect on the through life costs of a vessel. However, most computer-based ship design software focuses on the detailed phases of design related to structure and production. To investigate and develop innovative solutions, the designer requires a tool that does not enforce detailed definition and allows easy reconfiguration of arrangements and systems. In addition, many calculation exercises may be automated allowing the designer to spend more time focusing on the solution.

Spreadsheet solutions offer a partial solution allowing the designer to look at initial sizing. But as generic tools, the mathematical relationships are based more on historical rules entered by the designer than calculations on representative ship data structures. Often designers prefer begin directly with detailed design tools because only they can make representative analysis, accepting the overhead associated with detailed definition. To allow the definition of complex assemblies, detailed design tools introduced topological relationships allowing the definition of components to be based on others. This greatly improves design as changes can propagate through the ship model to update all related parts.

Paramarine offers an intermediate solution. The system provides integrated modules covering the different engineering disciplines associated with ship design in an environment in which allows the specification of parametric, mathematical and topological relationships. This allows the user to work with simpler representations of components but include representative calculations where necessary. The resulting solution is easier to manipulate and allows the designer to focus on identifying a configuration which satisfies the functional requirements.

Developing a parametric design solution is not a trivial task and many ship design tool provide this capability using scripting or structured data. However, designers, generally, do not have the time to use or understand these capabilities. In Paramarine, the Early Stage Design module provides a readymade framework to implement a parametric design. The framework provides an excellent opportunity to integrate parametric hull generation which, in the past, has been little use due the complexity of integrating this approach with other parts of the design tool.

The Early Stage Design Concept

The “Functional Building Block” methodology developed by University College London (UCL) (Andrews and Pawling [1]) offers an approach for managing the ship design process in the preliminary stages and has been incorporated into the Paramarine integrated ship and submarine design system. This approach originates from the design of naval vessels where there are a greater number of requirements and system integration more complex in comparison with most commercial vessels. In addition, in demonstrating that a design meets requirements it can be a useful tool for satisfying a potential customer, especially in the case of naval vessels, where a large public expenditure is involved. The approach can be applied just as equally to commercial vessels allowing outline arrangements to be flexibly evaluated in cases where an innovative solution may dependent on the compact integration of different systems or if the vessel configuration is unfamiliar, unusual or novel in respect to the designer’s experience.

The Building Block methodology appears, initially**,** more abstract than other common approaches used to develop ships. The designer approaches the problem in terms of *functional* issues in order to develop the *form* of the solution. This fosters a much closer relationship between the design definition and requirements making potential solutions easier to evaluate. The definition itself consists of a clear hierarchy of elements divided up into functional groups, for example, in the case of a naval vessel, would represent the float, move and fight etc, aspects of the requirements. These groups are then further subdivided into the individual elements (Building Blocks) representing the components and systems of the vessel that will address the requirements in each of these areas. Block elements without sub-blocks represent an individual component of a design based on the element's attribute data. Attribute data encapsulates the geometric, functional and topological characteristics so that the role of Block element within the design is clearly defined. The functional characteristics of a block are initial the most important defining whether it addresses requirements by providing (supplies) capability and/or if it requires (demands) some capability to be provided to function. Functional characteristics may consider:

- Weight/Buoyancy
- Space (area, volume)
- Consumables (water, fuel, materials etc)
- Services (electricity, HVAC, high pressure air etc)
- Personnel (unskilled, skilled, responsible etc)
- User defined

Geometric attributes model the extents of the component and may be simulated by simple primitive bodies although, in the case of equipment, visualisation of the actual shape is more useful. Topological attributes may define relationships the block may have to others in the hierarchy allowing inter-block proximity or infringements to be identified or design of any service/consumable infrastructure to be taken into account.

As the design is being developed, the complete hierarchy can evaluated to present the accumulated characteristics and identify any conflicts between components. Geometrically, the extents of the hierarchy will identify the overall sizing of the vessel and the weight and centroid will identify the volumetric characteristics of a required hull form, which can subsequently be included as a block supplying buoyancy once the hull geometry is available.

The Building Block hierarchy provides the means to define the constituent components parts of the design and can be analysed to identify shortfalls in characteristics and conflicts. Having defined the parts, the designer should be able to move individual blocks, searching for the best configuration which provides a balanced design. Implemented as part of an integrated ship design tool, a 'drag and drop' approach can be used to manipulate the 'blocks' and analysis functions invoked automatically after each interaction. Consequently, the design is in a state where is can be easily changed and accept any changes to requirements that may occur during a project.

Potentially, parametric design can also offer great benefits in the definition environment. It one of the few areas where hull form generation may be used effectively to automatically produce a surface supplying all the characteristics required to support the design. The Building Blocks methodology has been implemented within Paramarine, a ship design system which promotes the use of the parametric approach in ship design.

Paramarine

Paramarine is an integrated ship design environment developed as a replacement ship and submarine design toolset for the MoD Goddess system. The system has been developed afresh from the bottom up, embracing modern object-orientated software development techniques and is based upon the Parasolid solid modelling library. As would be expected of any modern integrated ship design environment standard modules cover the many of the familiar engineering disciplines such as stability, powering and structures etc, to name but a few.

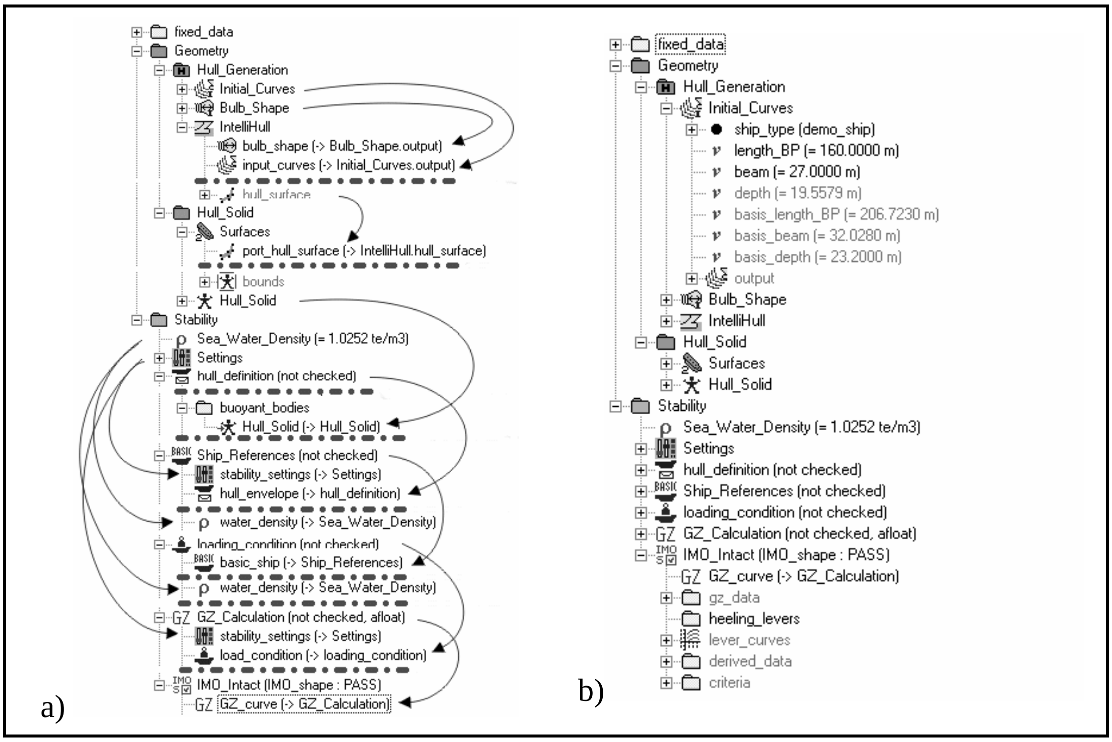


Figure 1: References (pointers) are used to build and connect aspects of the design together (a). Consequently, significant changes in the design can be investigated easily changing parameters (b).

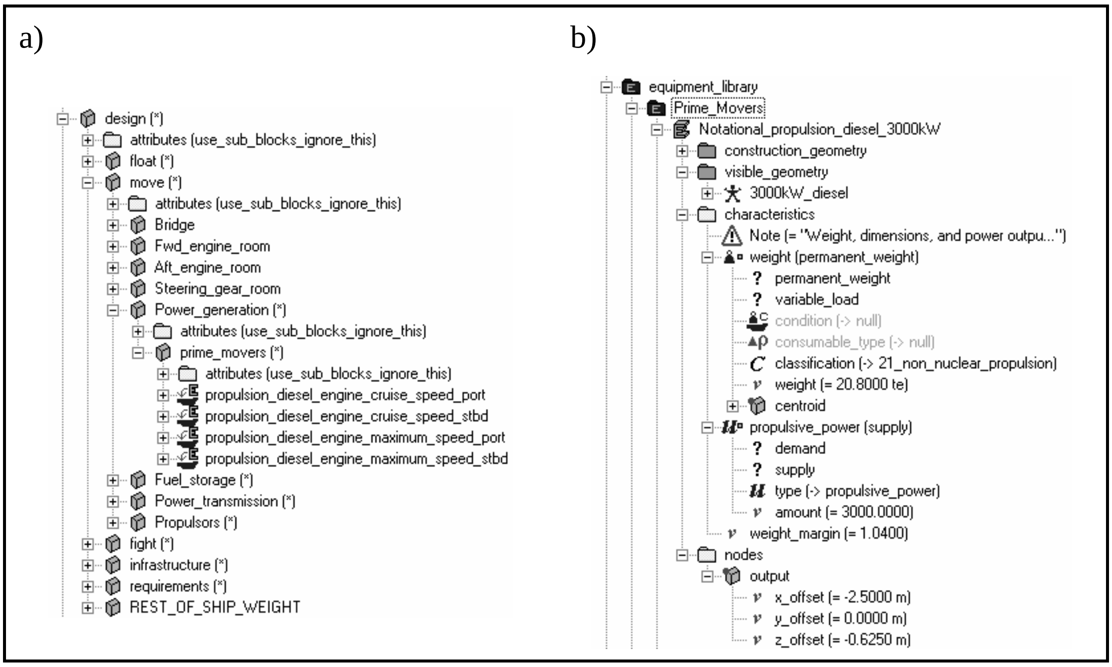
However, unlike most current systems the design environment itself is based on an object-orientated framework where users are directly exposed to the full data structure allowing parametric connection of all aspects of both the product model and analysis together. Objects make up the individual definition elements of a design usually with a set of corresponding attributes which may be numerical values, other objects or references, Figure 1a. The design itself will consist of a collection of simpler objects assembled together to form a more complicated design. Therefore, the topological aspects are modelled as accurately as the geometric aspects.

One of the key objectives of Paramarine is the management of the design configuration. In essence, the system will attempt to maintain every aspect of the design in a correct and updated state by responding to user changes. This not only means that out of date elements are recalculated but also that any connections (references) are legitimate, for example, ensuring that a connection points to the right kind of object (type-checking) or that the dimensional units of a formula equate correctly taking into account the input parameters and the expected outcome.

With fairly limited experience any user can quickly define and arrange a set of objects which can be used to investigate various characteristics of a design. In Figure 1b, the connective details between the stability assessment and hull definition have been collapsed to reveal the underlying link between the length between perpendiculars and breath parameters. Following this approach, and with good planning, a complete design with a range of characteristics can be developed with modification and hence any investigation driven by a small parametric definition.

Early Stage Design Implementation

The Paramarine early stage design module is based upon the Building Blocks methodology and implements it in the tool's unique definition style. The tree definition structure is capable of representing the hierarchy of building blocks perfectly, Figure 2a. Individual blocks, Figure 2b, can be assigned visualisation geometry and functional characteristics.



```
design (*)
  attributes (use_sub_blocks_ignore_this)
  float (*)
  move (*)
    attributes (use_sub_blocks_ignore_this)
    Bridge
    Fwd_engine_room
    Aft_engine_room
    Steering_gear_room
    Power_generation (*)
      attributes (use_sub_blocks_ignore_this)
      prime_movers (*)
        attributes (use_sub_blocks_ignore_this)
        propulsion_diesel_engine_cruise_speed_port
        propulsion_diesel_engine_cruise_speed_stbd
        propulsion_diesel_engine_maximum_speed_port
        propulsion_diesel_engine_maximum_speed_stbd
    Fuel_storage (*)
    Power_transmission (*)
    Propulsors (*)
  fight (*)
  infrastructure (*)
  requirements (*)
  REST_OF_SHIP_WEIGHT
```

```
equipment_library
  Prime_Movers
    Notational_propulsion_diesel_3000kW
      construction_geometry
      visible_geometry
        3000kW_diesel
      characteristics
        Note (= "Weight, dimensions, and power outpu...")
        weight (permanent_weight)
          ? permanent_weight
          ? variable_load
          condition (-> null)
          consumable_type (-> null)
          C classification (-> 21_non_nuclear_propulsion)
          v weight (= 20.8000 te)
          centroid
        propulsive_power (supply)
          ? demand
          ? supply
          type (-> propulsive_power)
          v amount (= 3000.0000)
        v weight_margin (= 1.0400)
      nodes
        output
          v x_offset (= -2.5000 m)
          v y_offset (= 0.0000 m)
          v z_offset (= -0.6250 m)
```

Figure 2: The functional building block hierarchy (a) contains individual building blocks (b) which geometric and functional characteristics, e.g. a 3000kW diesel with visualisation, weight, power and connectivity (nodes) attributes.

Once defined, individual building block elements can be interactively moved around either numerically or using the mouse to identify arrangements with few infringements and satisfy requirements, Figure 3.

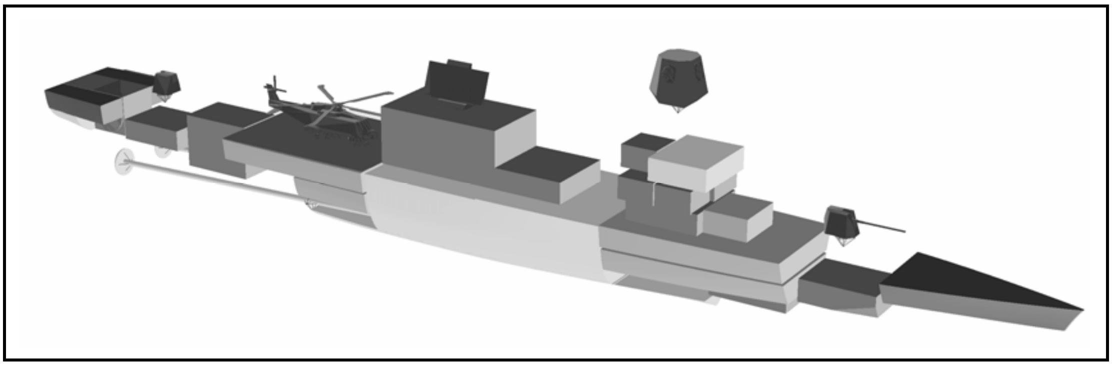
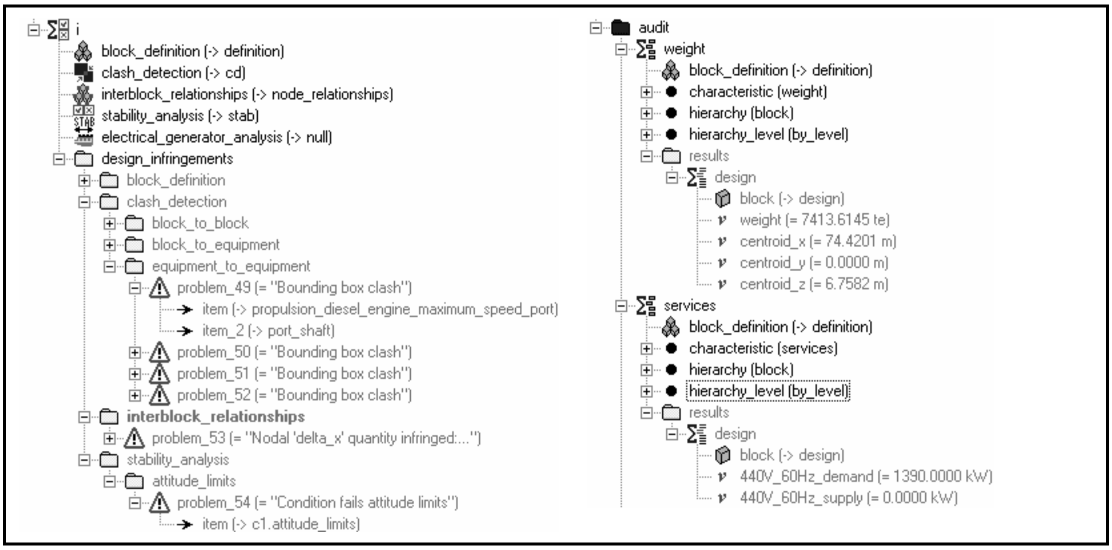


Figure 3: A coarse layout of Building Blocks for a frigate design including equipment geometries.

The performance of each arrangement is reviewed by looking at the analysis and audit of the building blocks hierarchy. These "views", Figure 4, provide two alternative ways at looking at the composition of the ship system. The infringements view (left) highlights areas where there are conflicts in the design, in the building blocks definition and in higher level analysis such as stability. Locations where the requirements have not been are flagged such as under supply or equipment intersections are highlighted. Auditing (right) can be used to understand the efficiency of the design ensure that certain characteristics are not over supplied.



```
⊟ i
    block_definition (-> definition)
    clash_detection (-> cd)
    interblock_relationships (-> node_relationships)
    stability_analysis (-> stab)
    electrical_generator_analysis (-> null)
  ⊟ design_infringements
    ⊞ block_definition
    ⊟ clash_detection
      ⊞ block_to_block
      ⊞ block_to_equipment
      ⊟ equipment_to_equipment
        ⊟ problem_49 (= "Bounding box clash")
            → item (-> propulsion_diesel_engine_maximum_speed_port)
            → item_2 (-> port_shaft)
        ⊞ problem_50 (= "Bounding box clash")
        ⊞ problem_51 (= "Bounding box clash")
        ⊞ problem_52 (= "Bounding box clash")
    ⊟ interblock_relationships
      ⊞ problem_53 (= "Nodal 'delta_x' quantity infringed:...")
    ⊟ stability_analysis
      ⊟ attitude_limits
        ⊟ problem_54 (= "Condition fails attitude limits")
            → item (-> c1.attitude_limits)
```

```
⊟ audit
  ⊟ weight
      block_definition (-> definition)
    ⊞ ● characteristic (weight)
    ⊞ ● hierarchy (block)
    ⊞ ● hierarchy_level (by_level)
    ⊟ results
      ⊟ design
          block (-> design)
          v weight (= 7413.6145 te)
          v centroid_x (= 74.4201 m)
          v centroid_y (= 0.0000 m)
          v centroid_z (= 6.7582 m)
  ⊟ services
      block_definition (-> definition)
    ⊞ ● characteristic (services)
    ⊞ ● hierarchy (block)
    ⊞ ● hierarchy_level (by_level)
    ⊟ results
      ⊟ design
          block (-> design)
          v 440V_60Hz_demand (= 1390.0000 kW)
          v 440V_60Hz_supply (= 0.0000 kW)
```

Figure 4: Building Block definition can be evaluated in terms of infringements (left) or audited (right) to review the balance between demand and supply of the functional characteristics.

Parametric Hull Generation

Parametric hull generation techniques are mathematical procedures which create the hull surface usually from a set of numerical parameters. These techniques became a popular research area with the introduction of affordable computing technology. Early techniques offered promise but were often limited by the capabilities of the mathematical techniques that could used to represent the shape of the hull as a surface or sections. Interest in these methods reduced with the successful adoption of parametric curves and surfaces, such as NURBS, into ship design tools. However, interest in parametric hull generation has again increased as quicker methods of obtaining a hull form are required to address ever shortening design phases.

The greatest shortcoming of the directly defined hull surfaces used in the majority of ship design tools is the effort required to effect systematic changes beyond simplistic transformations applied to the whole surface. Parametric hull generation offers a solution by producing the hull surface in respect of important form parameters. As concept design is mostly concerned with the identification of a set of characteristic form parameters for a particular style of hull, the loss of detailed definition capabilities is generally of no consequence and will allow the design process to focus on the important aspects of the solution. Modern parametric hull generation techniques have embraced NURBS allowing the resulting hull surface to be interoperable with detailed hull surface design tools.

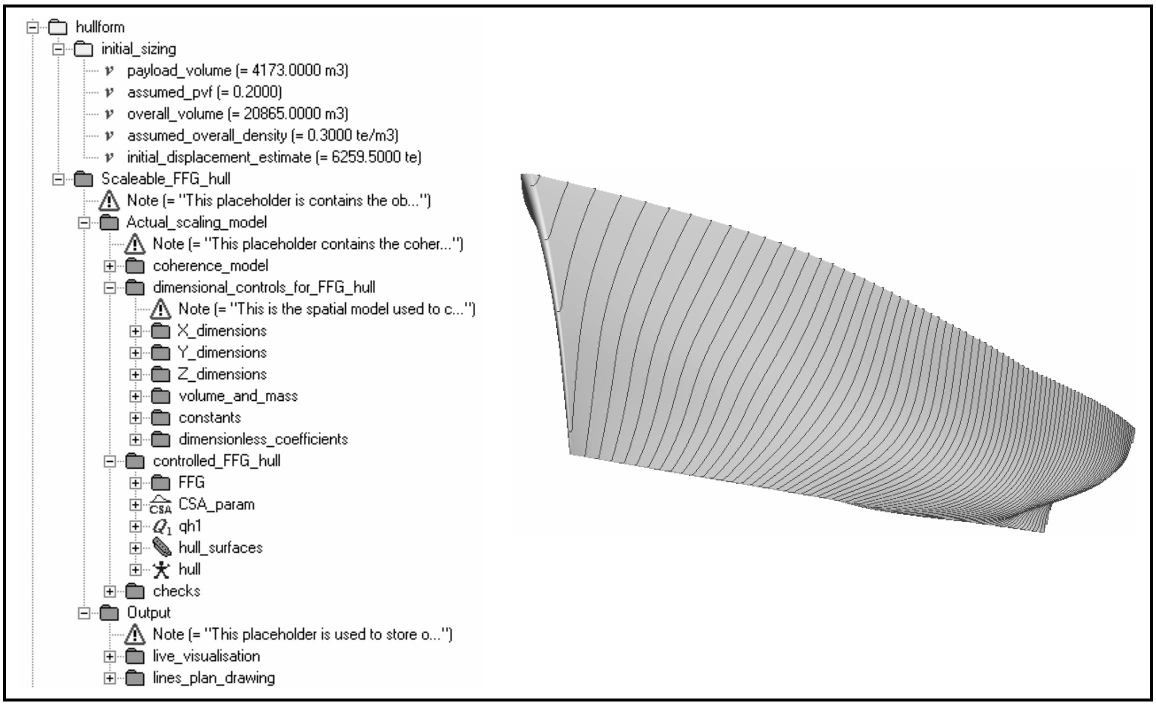
Although capable of producing surfaces with detailed features, experience has show than the most effective approach is to minimise the number of parameters that control the hull form. The primary reason for this is that in attempting to generate the hull surface there can sometimes be conflict between parameters resulting in a technique which is constrained to smaller range of variations. Furthermore, as all of the parameters must be specified to complete the definition, representative values must be found for features which may not be important or available early in the design process.

In Paramarine, parametric hull generation is focused toward the development of hull surfaces to accommodate ship layouts, in particular those produced by the Early Stage design module. Several approaches are available:

- QuickHull – Generation and modification of Frigate type hull forms.
- IntelliHull (Bole [2]) – Generation of conventional ship type hull forms.
- Hull Generator – A parametric hull surface design tool
- User developed – Hull forms can be developed by using the modelling features of Paramarine alone.

Quickhull

Quickhull is a technique for generating frigate hull forms. The surface it self is quite simple consisting of a single NURBS surface with a 5 x 8 point control polygon, but associated with it is range of design rules which manipulate the surface to form the correct shape from a small set of design parameters. The surface is initially controlled by a set of key points which are connected together using curves representing the profile, deck line, stem, transom and midship section and include the aft cut-up point. The hull surface is formed between these curves and using a target cross sectional area curve (CSA), the internal control points are moved longitudinally with the objective of matching the CSA of the hull surface to the target curve as closely as possible.



- hullform
  - initial_sizing
    - payload_volume (= 4173.0000 m3)
    - assumed_pvf (= 0.2000)
    - overall_volume (= 20865.0000 m3)
    - assumed_overall_density (= 0.3000 te/m3)
    - initial_displacement_estimate (= 6259.5000 te)
  - Scaleable_FFG_hull
    - Note (= "This placeholder is contains the ob...")
    - Actual_scaling_model
      - Note (= "This placeholder contains the coher...")
      - coherence_model
      - dimensional_controls_for_FFG_hull
        - Note (= "This is the spatial model used to c...")
        - X_dimensions
        - Y_dimensions
        - Z_dimensions
        - volume_and_mass
        - constants
        - dimensionless_coefficients
      - controlled_FFG_hull
        - FFG
        - CSA_param
        - qh1
        - hull_surfaces
        - hull
      - checks
    - Output
      - Note (= "This placeholder is used to store o...")
      - live_visualisation
      - lines_plan_drawing

Figure 5: Quickhull: Definition objects and coherency model (left) are used to produce the hull representation (right) for a basic set of parameters.

While the process of generating a Quickhull surface is fairly straightforward, selecting a set of parameters (key points and CSA curve) that will develop a satisfactory surface is difficult as the input parameters are not independent. A simple solution is to base a design upon a parent hull form and distort the shape to fit the new design. However, a parent design is not always available. An alternative approach is the use of the coherency model developed by UCL for use with the Early State Design module. A coherency model implements and extends a set of design rules for monohull warship geometry (van Griethuysen [3]) to produce a set of input parameters for the Quickhull generator based on displacement, waterline length and circular 'M' additionally allowing some customisation of various ratios and coefficients. The objective of this approach is that given a basic displacement and size provided by the Early Stage Design module, a frigate type hull form can be generated to accommodate the devised arrangement. The coherency model is entirely defined using Paramarine definition element with only the iterative modification of the hull surface to match the target CSA being performed by internal functions.

IntelliHull

As the hull design capabilities provided by most integrated ship design systems are orientated towards detailed consideration, there is a gap in technology for a hull design tool which could rapidly react to the changes in geometry required for concept and initial design. Any body experienced in entering hull forms into ship design tools will have recognised that, for convention forms at least, the same pattern of definition data is often used. This highlights that although there may be variation in shapes and dimensions, there is commonality in the underlying arrangement or topology of these shapes and is a factor which could be used to assist in the initial development and change of the hull surface definition.

It is quite likely that the designer has a good mental picture of the form before developing a hull surface and that a concept design tool should provide the tools to convert that image into a three dimensional representation. Fundamentally, the designer should be able to control numerical (quantifiable) characteristics parametrically and control shape (qualitative) characteristic through interactive manipulation. The tool would assist the designer by providing tools which can constrain geometrical definition in line with the form topology, modify (transform) the geometry in ways allowed by the topology only, and insert automatically generated geometry to fulfil the correct definition required to meet the form topology.

The IntelliHull technique was developed to demonstrate this concept for conventional ship hull forms using a single NURBS surface. The implementation separates the designer from the generated surface so that the tool can process the definition, constrain or add definition where necessary. A summary of the features is as follows:

- Designer manipulates transverse definition curves, through which the hull surface is blended/lofted.
- Definition curves can be constrained to model specific shapes (straight sections blends etc). Constraints can be based on the shape of other curves.
- Curves generated if not supplied, to produce parallel middle body and control hydrostatics in line with topology
- Surface operations such as bending and warping [4] can be applied to change the generated surface without affecting the definition curves. Hence, bulb shapes can be added without adversely increasing the amount of definition and the ease in which it can be modified. Subsequently, the definition of local hull features is independent of the definition of the main hull envelope.

The Paramarine implementation focuses on the parametric capabilities of IntelliHull providing a set of definition objects which can be used to generate an initial surface based on a set of stored templates in a matter of seconds that can be subsequently refined as part of the design process. This implementation differs slightly from the original by generating a hull surface which meets, as closely as possible, the dimensions specified by parameters. The original implementation uses the parameters to change the geometry which could be further edited by the user.

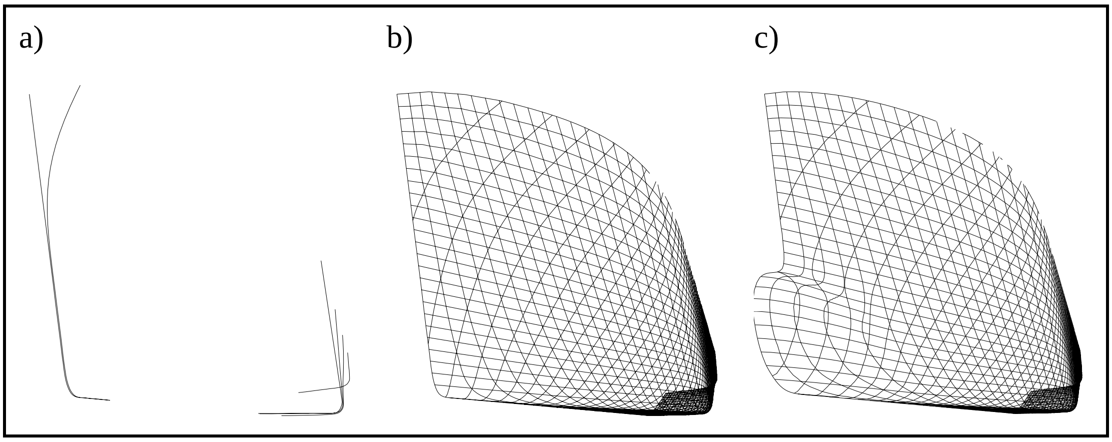
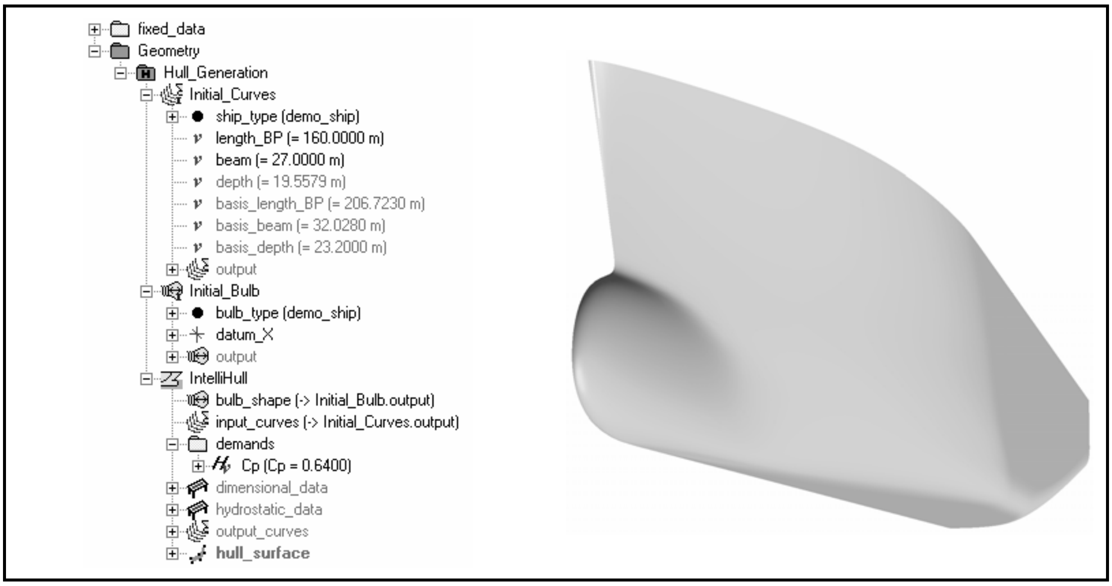


Figure 6: The three stages of defining an IntelliHull, a) initial curves, b) basic hull form and c) application of local features, e.g. Bulb.

The full version of IntelliHull is available in the author’s own tool PolyCAD [5] and development is underway with the intension of applying the many of the ideas to a multiple patch hull surface definition capable of working with generic hull form topology.



```
fixed_data
Geometry
  Hull_Generation
    Initial_Curves
      ship_type (demo_ship)
      length_BP (= 160.0000 m)
      beam (= 27.0000 m)
      depth (= 19.5579 m)
      basis_length_BP (= 206.7230 m)
      basis_beam (= 32.0280 m)
      basis_depth (= 23.2000 m)
      output
    Initial_Bulb
      bulb_type (demo_ship)
      datum_X
      output
    IntelliHull
      bulb_shape (-> Initial_Bulb.output)
      input_curves (-> Initial_Curves.output)
      demands
        Cp (Cp = 0.6400)
      dimensional_data
      hydrostatic_data
      output_curves
      hull_surface
```

Figure 7: The Paramarine implementation of IntelliHull. The definition objects are shown on the left and resulting hull is shown on the right.

Hull Generator

Hull Generator provides an alternative hull form definition technique based around topological characteristics rather than parametric generation. It follows the practice used in other tools by defining the hull as a patch network of NURBS surface patches. However, unlike other techniques, it offers the ability to define the hull form using “high level” definition in which the user expresses his or her wishes in a conceptually simple way and is not required to become immersed in the attributes or mathematics of b-spline entities.

Hull Generator is used to form the hull of a vessel using typically a small number of patch networks which can be attached together. The hull surface is refined by subdividing patches introducing further curves. Shape is controlled by modifying control vertices and associated attributes defining slope and continuity, Figure 8.

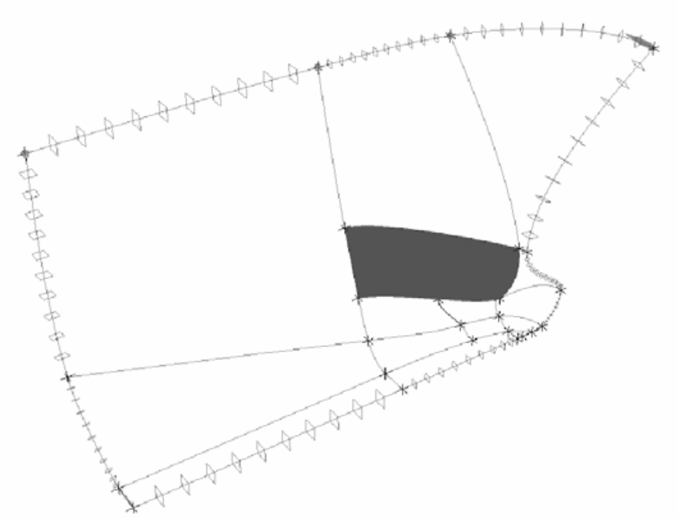
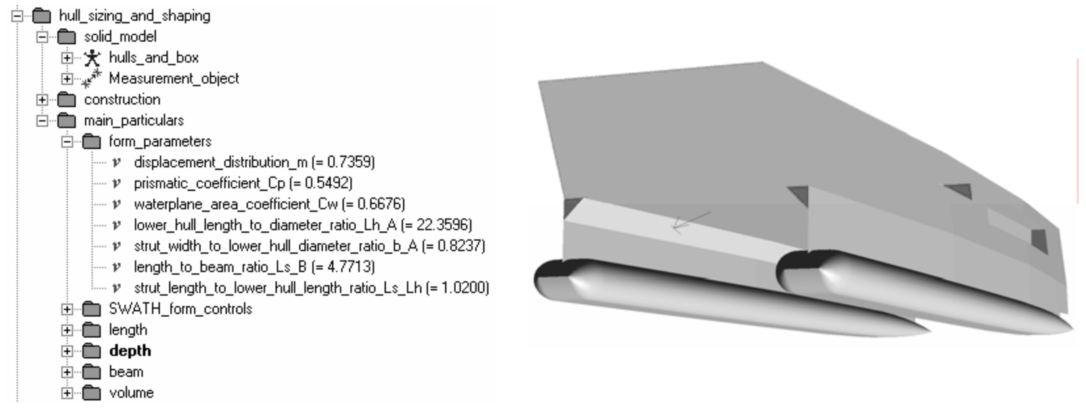


Figure 8: The points, curves and one patch from a network defining the bow region of a hull form. The shape controls for the upper deck line curve are also shown.

User Defined Parametric Hull Definitions



```
⊟ hull_sizing_and_shaping
  ⊟ solid_model
    ⊞ hulls_and_box
    ⊞ Measurement_object
  ⊞ construction
  ⊟ main_particulars
    ⊟ form_parameters
        v displacement_distribution_m (= 0.7359)
        v prismatic_coefficient_Cp (= 0.5492)
        v waterplane_area_coefficient_Cw (= 0.6676)
        v lower_hull_length_to_diameter_ratio_Lh_A (= 22.3596)
        v strut_width_to_lower_hull_diameter_ratio_b_A (= 0.8237)
        v length_to_beam_ratio_Ls_B (= 4.7713)
        v strut_length_to_lower_hull_length_ratio_Ls_Lh (= 1.0200)
    ⊞ SWATH_form_controls
    ⊞ length
    ⊞ depth
    ⊞ beam
    ⊞ volume
```

Figure 9: A SWATH vessel controlled by numerical parameters (left).

The Paramarine modelling environment is flexible enough to allow the development of user’s own parametric hull definitions. Solid modelling operations and surface control can be combined to create detailed hull forms. The modelling environment can provide volumetric characteristics by interrogating the solid definitions. Figure 9 shows a SWATH design generated from numerical parameters (left).

Composing an Integrated Design

The early stage design module does not enforce any particular level of detail and it can be useful to start with the coarsest of models. Components can be defined using simple primitives and moved around to review various configurations. Defining the characteristics associated with individual blocks will allow the framework to collate information within the design and feed back the accumulated properties at the top level. Characteristics such as weight and buoyancy can be used to access the floating position and stability characteristics with a representative hull form, Figure 10. The arrangement can easily reconfigured by dragging the blocks around and top level characteristics analysed by the design software automatically.

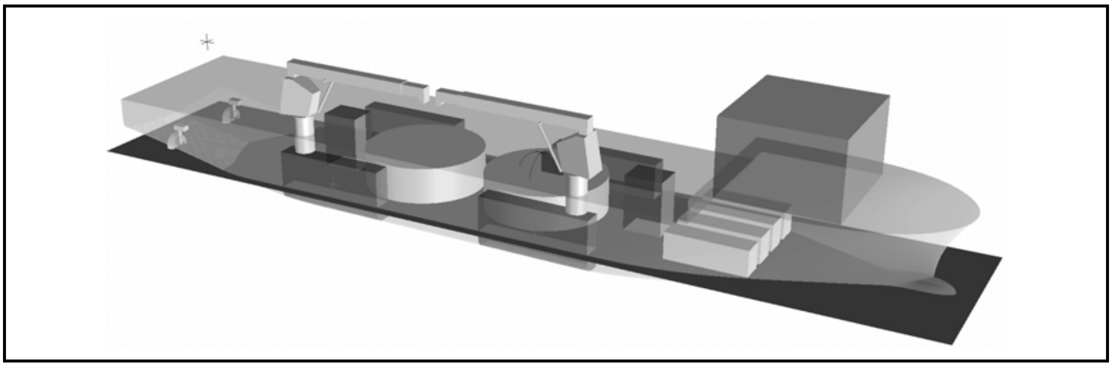


Figure 10: A coarse early stage design of a supply vessel outfitted with cable reels, cranes and propulsion to review an initial configuration.

Once the initial configuration has been settled the compartment configuration can be introduced, Figure 3. At this stage, the hull form can be parametrically connected using information from the extents and the weight of the building block definition. The initial functional block definition can be developed in more detail to address the operational ship systems and related connectivity’s which can be used to assess feasibility in terms of proximity, complexity and clashes with other systems. The detailed version will allow additional implicit requirements to be introduced as supporting systems will introduce their own.

Potentially the design can become relatively complex, although definition information remains manageable. It is possible to be detailed in some areas if they are critical to the functionality of the ship, while leaving others undefined. The example in figure 11 shows a replenishment support ship where the functionality of the deck equipment is important. The space requirements and inter-relationships of the helicopter deck and cargo handling system can be modelled to establish the space required for these two areas.

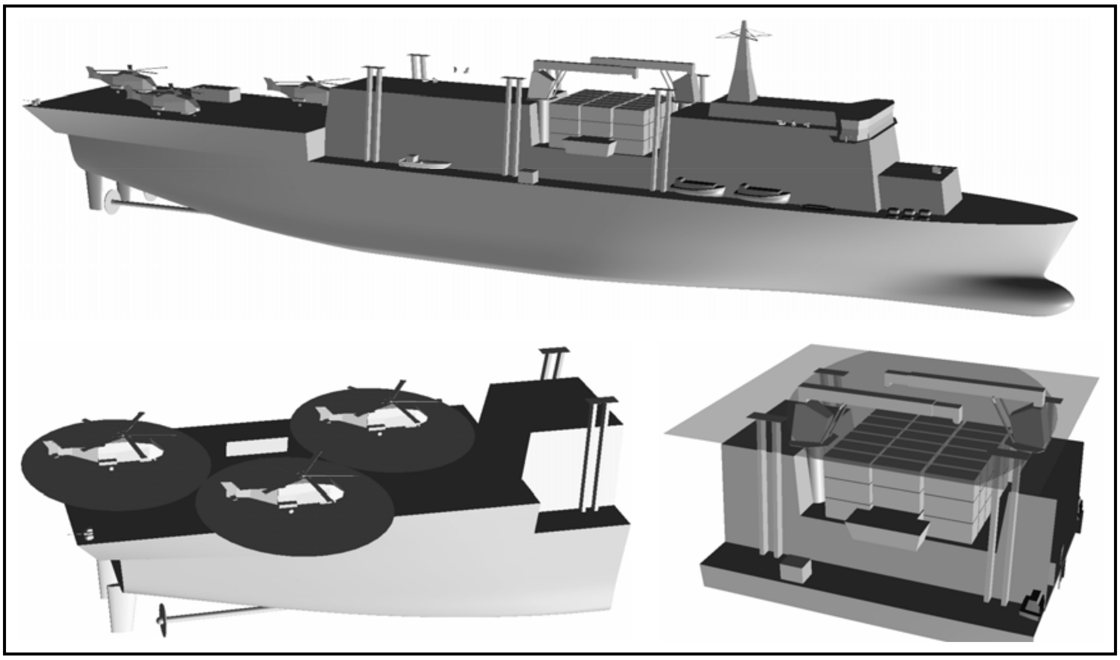


Figure 11: A replenishment support ship where the design concentrate on the deck systems leaving internals and the hull form to be managed by the parametric relationships.

The Next Step: Design for Production (DFP)

Following on from the Early Stage Design, the Design for Production module [6] takes a similarly detailed look at the production characteristics of the early stage design model. With a suitable structural definition, the module can be used to break the design down into production blocks and their constituent parts including the systems and outfitting to review the composition of the design in terms of materials and processes. Modelling the production activities at the component level leads to realistic estimation of costs and identification of production infringements.

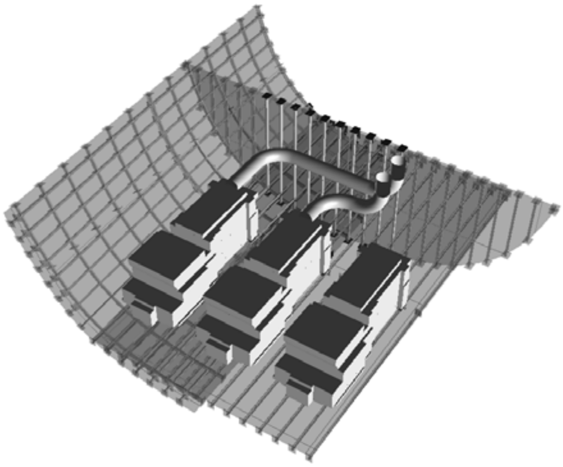


Figure 12: A DFP model illustrating structural and outfit components.

Conclusion

Paramarine’s Early Stage Design module provides a hierarchical framework for parametrically modelling the design of a ship at the functional level. The designer can use a relatively simple geometric model to represent components with a range of difference characteristics (Building Blocks). The hierarchy is evaluated to assess the accumulated characteristics establishing shortfalls, infringements and inconsistencies, highlighting these facts to the design. The building blocks can be interactively manipulated to assess different design configurations.

The Paramarine environment is well suited to the use of parametric hull generation techniques. Information collated by the Early Stage design module can be used to generate a hull surface which is effectively “wrapped” around the building block definition. Several different hull generation methods are available but the tool is flexible enough for user to implement their own formulations.

Where system integration is important to a vessel’s functionality, the Early Stage Design module provides a unique solution to the design problem allowing the designer to flexibility assess different configurations and make representative analysis without the need for detailed definition.

References

1. Andrews. D., Pawling. R., SURFCON – A 21st Century Ship Design Tool, University College London, 2002.
2. Bole. M., A Hull Surface Definition Solution based on Form Topology and geometric Constraints, PhD Thesis, University of Strathclyde, 2002.
3. van Griethuysen, W. J., On The Choice of Monohull Warship Geometry – A Technique for optimising hull forms for seakeeping, high speed resistance and fuel consumption performance, RINA, 1993.
4. Piegl. L, Tiller. W, The NURBS Book, Springer, 1995.
5. http://www.polycad.co.uk
6. Muñoz, J. A., Forrest C. J. M., Advantages of Software Integration from Initial Design through to Production Design, ICCAS, 2002.