WHITEPAPER

FRIENDSHIP SYSTEMS

PRACTICAL SHAPE OPTIMIZATION USING CFD

Stefan Harries, FRIENDSHIP SYSTEMS
February 2015 (Version 1.1)

KEY WORDS

Simulation-driven design, simulation-based design, Computer Aided Engineering (CAE), Computer Aided Design (CAD), upfront CAD, Computational Fluid Dynamics (CFD), design studies, formal optimization, parametric modeling, adjoint simulation, virtual prototypes, engineering models, design process, product development, functional surfaces, flow-exposed shapes, variable geometry, simulation-ready CAD

1. INTRODUCTION

Optimization is a human trait. People want to find a good, possibly the best solution under the given circumstances. Getting the nicest vacation within a certain budget is essentially not so different to designing a pump with smallest pressure loss, a ship with lowest energy consumption or an offshore platform with the least risk of damage in rough seas: Mathematically speaking it is minimizing (or maximizing) one or several objectives within a set of constraints.

The term optimization is used ambiguously though. Traditionally, an engineer would rightfully say that something was optimized once a handful of feasible options had been considered. However, since a few years, modeling tools, formal optimization algorithms, simulation codes and adequate computer power have become available that allow the generation and assessment of hundreds and even thousands of virtual prototypes. Today, these resources are not only accessible to a few dedicated experts in large companies, but are increasingly utilized by designers even in small companies.

The techniques involved are manifold and, to some extent, domain dependent. The most important watershed is if optimizations are performed with respect to fluid dynamics or structures. In this paper, focus is put on shape optimization for aero- and hydrodynamic performance. Prominent examples are the internal thermo-fluid dynamics of compressors and turbines, the external aerodynamics of cars, the hydrodynamics of ship hulls and their propulsion systems as well as the fluid dynamics of mixers, valves, ducts and pumps.

This paper aims at giving an overview so that the reader would come to know common techniques and typical applications, illustrated along projects realized within the Upfront CAE system CAESES® (www.caeses.com). An encompassing treatment would call for a book rather than a paper. Consequently, many interesting developments and details are omitted here for the sake of an informative, yet reasonably light read. The paper addresses engineers facing optimization projects, managers wishing to get a general understanding, and students developing an interest in the field of shape optimization using CFD.

2. BENEFITS

Shape optimization offers several benefits: It increases a company's competiveness via

- Better understanding of the design task (and the design space),
- Creating products with superior performance (and better trade-offs),
- Allowing shorter time-to-market (and faster response to market changes),
- Reducing risk (and building confidence),
- Saving costs (and avoiding expensive late changes).

Shape optimization is conducted both for investigating new ideas and possibilities at the initial design stage and for fine-tuning of a given product at a later stage when only small changes are still acceptable, Fig.1.

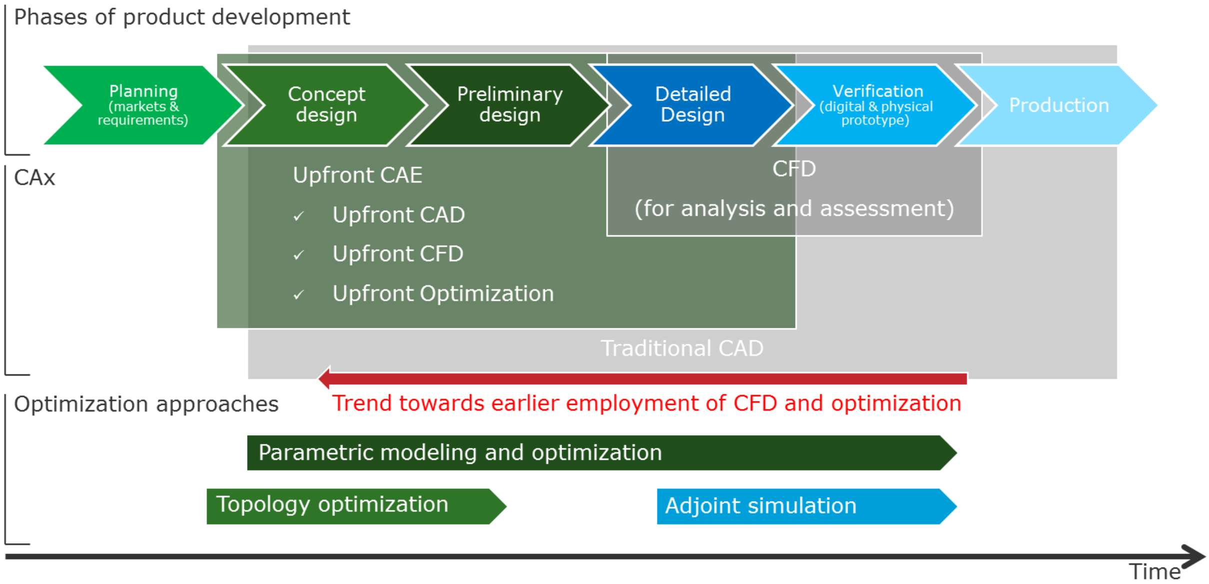

Fig. 1: Phases of product development

Naturally, the potential for major gains is highest the earlier an optimization is undertaken. Even experienced design teams can find interesting new ideas during an initial design phase when running design studies and optimizations. Furthermore, if little experience is available, say a brand new product shall be developed, optimization helps people gain insight.

With flow-exposed surfaces that feature compound curvature small yet concerted shape adjustments bring about tangible improvements. For example, automated optimization of ship hulls typically yield a three to five percent gain in energy efficiency even when starting from a mature baseline (Harries, 2008).

It is worthwhile to point out that optimization seems to be mainly about finding products of superior performance. It is of equal importance, however, that optimization is also about providing options from which to choose, complementing the knowledge base for high-quality decisions.

3. OPTIMIZATION APPROACHES

There are two major approaches: Parameter-based and parameter-free shape optimization. Parameter-based optimization, as the name suggests, is built on parametric models. Parameter-free optimization comprises topology optimization and adjoint simulation.

3.1 PARAMETER-BASED OPTIMIZATION

Parametric modeling is the definition of a product (or the representation of system behavior) by means of important descriptors. Today, most optimization projects are parameter-based. The main reasons are that, firstly, design teams can easily interpret the meaning and impact of design parameters and, secondly, that multi-objective and multi-disciplinary optimizations can be carried out without any conceptual hurdle.

Many parametric models are developed within traditional CAD systems during the detailed design phase, Fig.1. These models are mainly intended for production and often contain details that complicate simulation (e.g. need for defeaturing). Thus, for shape optimization using CFD special parametric models are needed, so-called engineering models, which describe the product with as few significant parameters as possible, sometimes deliberately leaving out characteristics of lesser importance. These models address the concept and the preliminary design phases, focusing on simulation-ready CAD, and are realized within upfront CAD systems, Fig.1. Two major traits of upfront CAD are distinguished: Fully-parametric modeling and partially-parametric modeling, see (Abt and Harries, 2007) and (Harries et al., 2004).

Fully-parametric modeling

In fully-parametric modeling the entire shape is defined by means of parameters. A hierarchical model is created in which parameters describe all features of the envisioned product. Some parameters may be at a high level like the length, width and height of an object. Other parameters may determine details like an entrance angle at a particular location. Typically, many parameters are set relative to or as combinations of other parameters while some parameters may even be determined from additional analysis (e.g. formulas, background computation to reach a target value). A parametric model can be looked at as a system that takes parameters as input and produces a shape as an output. Any shape is realized from scratch and variants are brought about by simply changing the values of one or several inputs. For optimization, fully-parametric modeling is very powerful since it enables both large changes in the early design phase and small adjustments when fine-tuning at a later point in time.

While many traditional high-end CAD systems support or even advocate fully-parametric modeling, very few systems were actually developed for the parametrics of flow-exposed shapes such as turbine blades, ship hulls and pump volutes. These shapes often feature one distinct path in which design information changes rather slowly while the building strategy orthogonal to that path stays pretty much the same, Fig.2. For instance, the blades of a propeller do not change significantly from hub to tip. Rather, the (cylindrical) profiles are nicely defined by the same parameter set with different values at each radius, comprising chord length, pitch angle, maximum thickness and camber etc. Similarly, the sections of a ship hull change only gradually when sweeping from stern to stem, smoothly transferring between a handful of topologically different regions. Here, the parameter sets in longitudinal direction contain information about the deck, design waterline, sectional area curve, lateral profile etc. (Harries, 2010). Furthermore, the cross section of a volute does not really look that different from one angle to the next but rather evolves slowly in circumferential direction with input such as the area-to-radius distribution.

This is illustrated in Fig.2, taking models created in CAESES®. The system offers upfront CAD functionality dedicated to variable geometry as needed for optimization. High-level geometric constraints can be readily incorporated while processing the shapes for maximum fairness. The blade of a turbine or a propeller may have to comply with a prescribed area distribution (e.g. for load considerations) while the hull of a ship or an offshore platform must meet a given displacement (e.g. to comply with weight estimates). Rather than treating these characteristics as output from the modeling process and subsequently adjusting geometry manually until all constraints are satisfied, as would be done in most CAD tools, many constraints can be directly incorporated into the model. Fig.3 illustrates this for three variants of a semi-submersible (i.e., the lower part of an offshore platform). The parametric model is engineered such that a specified target volume is intrinsically satisfied for all modifications.

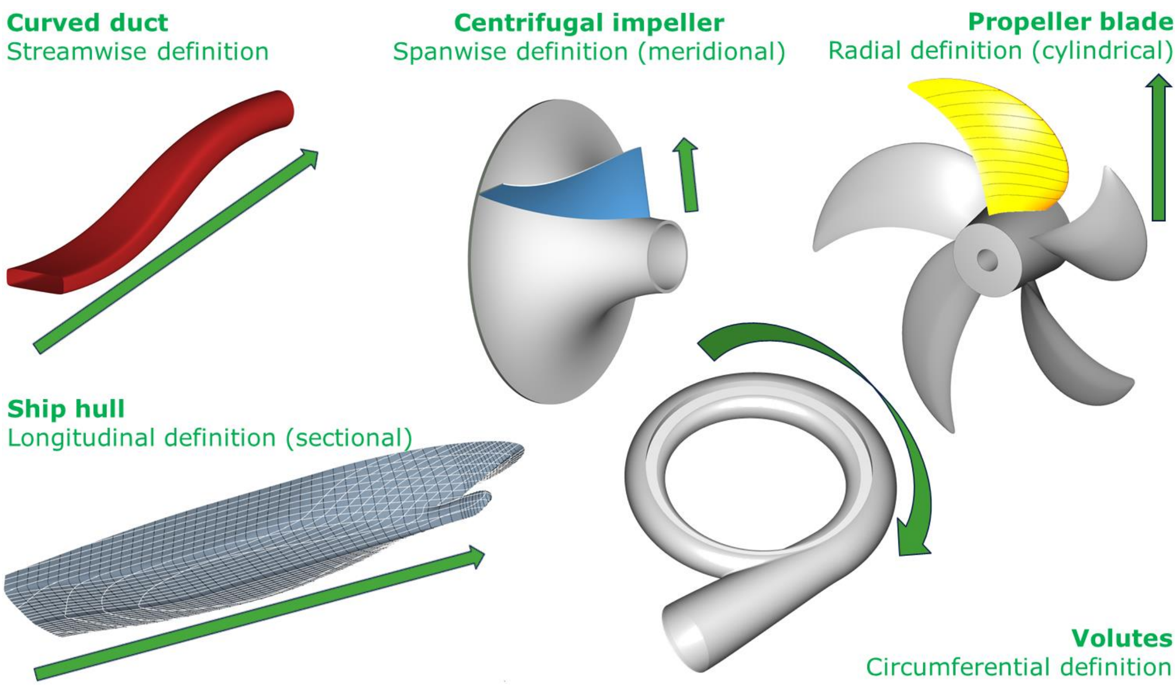

Fig. 2: Fully-parametric models with distinct directions of information
[See also www.caeses.com/support/videos]

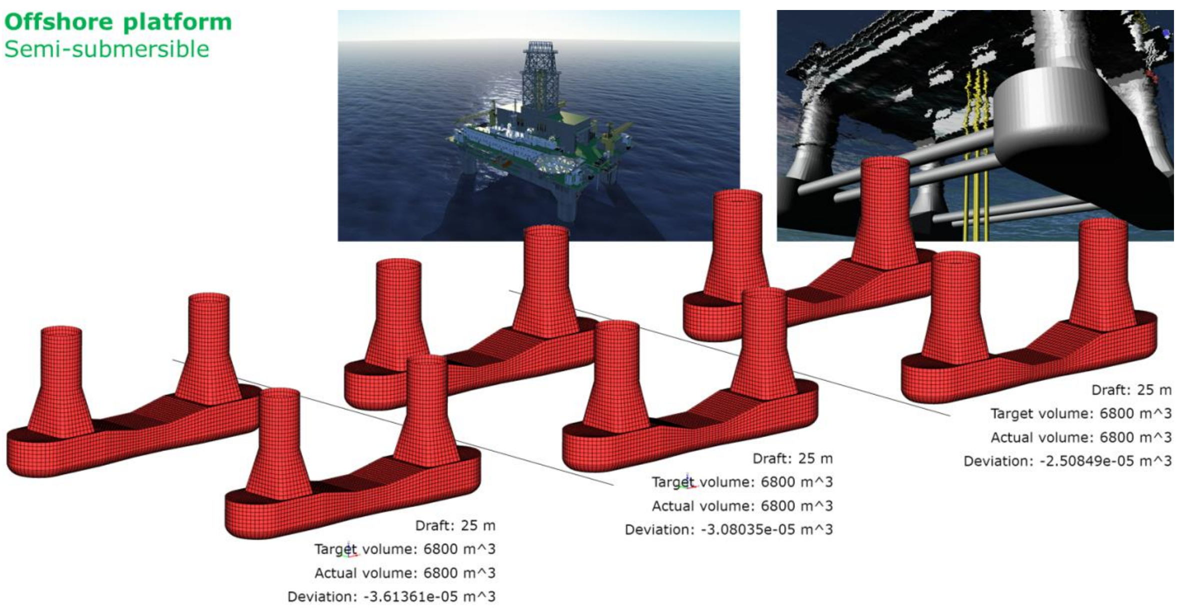

Fig. 3: Fully-parametric model of a semi-submersible with integrated constraint on volume at fixed draft

Partially-parametric modeling

In partially-parametric modeling only the changes to an existing shape are defined by parameters while the baseline is taken as input. The baseline may stem from any previous modeling process. Prominent representatives of partially-parametric modeling are morphing, free-form deformation and shift transformation (e.g. shifts in coordinate direction, radial shifts).

In morphing two or more objects are combined that are geometrically different but topologically identical. A well-known example is that of a cat and a dog which both feature two ears, two eyes and a snout (same topology) but, naturally, look quite different (different geometry). Mixing them with weights between zero and one creates anything from a pure cat to a pure dog with stages of transition from more-cat-than-dog to more-dog-than-cat. (In small-talk often all partially-parametric modeling techniques are subsumed as morphing even though that is not quite right from a mathematical point of view.)

In free-form deformation, also known as box deformation, the geometry to be modified is enclosed by a regular grid of vertices (i.e., rows, columns and layers defining a B-spline volume). For all parts of the initial shape which lie within the box (or lattice) a coordinate triple can be determined. By moving any of the vertices the box changes its shape and, along with it, the baseline is transformed (Sederberg and Parry, 1986). Here, the free coordinates of the box vertices serve as parameters. The technique is applicable to both surfaces and volumes, allowing box deformation to be exercised on a CFD mesh, too.

Shift transformations typically change any point in space by adding a certain displacement depending on the point's initial position. It can be applied to both continuous data (e.g. surface patches) and discrete data (e.g. points, offsets, tri-meshes as used for data exchange via STL). Fig.4 gives an example realized in CAESES®, showing a vertical shift of a container ship's bulbous bow.

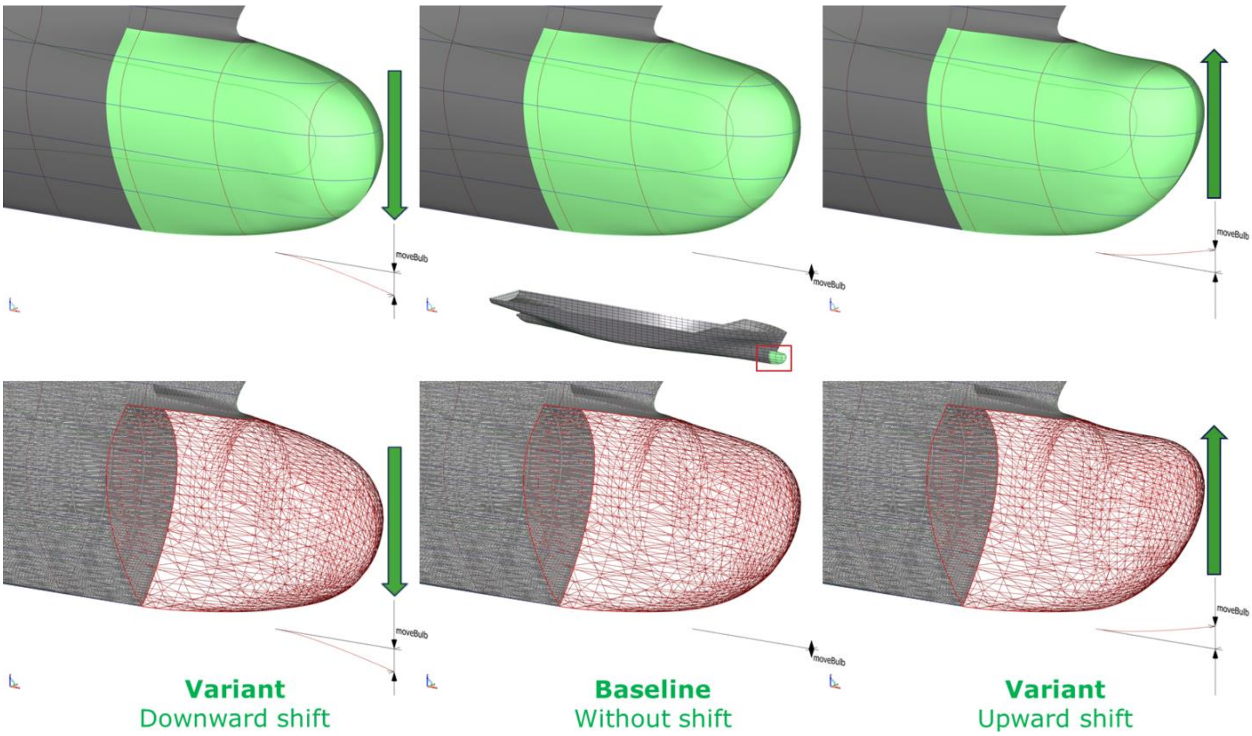

Fig. 4: Partially-parametric model for a downward vertical shift (left column), the baseline and an upward vertical shift (right column) applied to both B-spline surface patches and tri-meshes [center shows entire ship hull along with zoomed-in part (rectangle)]

Partially-parametric models are usually quick and fairly easy to set up. When compared to fully-parametric models they typically contain less knowledge (intelligence) about the product. In general, it is more difficult to excite large (game changing) modifications. After all, the new shapes are derived from the baseline and, thus, cannot look totally different. Still, they are well suited for fine-tuning without much overhead.

Parameter-based optimizations usually follow a simple line of action: All parameters that are believed to be important and that are under control of the design team are changed by an optimization algorithm, creating variants for assessment. For each variant one or several simulations are undertaken, returning objectives and constraints. This is repeated until either a certain number of variants has been studied (e.g. via a Design-of-Experiment) or a meaningful improvement was found (or, possibly, time and budget are consumed), see section 4 for details.

3.2 PARAMETER-FREE OPTIMIZATION

Topology optimization is mostly applied for solving internal flow problems early in the design process (conceptual level), Fig.1. The available domain along with its inflow and outflow boundaries is prescribed by the designer. The most favorable flow path is then automatically established for objectives such as pressure drop, homogeneity, lift and drag. The available space silts up (sanding) during the flow simulation, iteratively establishing the best organic shape (Stephan et al., 2008). Next, the shape has to be manually approximated in a CAD model, taking production constraints into account. At the end, CFD simulations are performed for the final shape to validate the outcome.

Adjoint simulations are utilized to fine-tune well advanced designs rather late in the process, Fig.1. The method is built on a CFD simulation for the baseline (primal solution), mostly solving the RANS equations, plus an additional simulation of similar effort that solves the so-called adjoint equations for the objective of interest (Stück et al., 2011). A CAD model is not necessarily involved at this stage since both the primal and the adjoint solution are run on the same computational grid. The adjoint simulation provides sensitivities on the domain boundaries, showing where to push geometry inwards and where to pull it outwards for improvements, Fig.5. Standard objectives are drag, homogeneity and pressure drop. A transfer needs to be made from the sensitivities towards the new shape. At CFD level this is done by shifting selected grid points on those domain boundaries that are free to change. To do so the grid points are moved by a small amount proportional to the adjoint sensitivities in the direction normal to their corresponding surfaces, displacing adjacent grid points in the discretized domain. Alternatively, a designer modifies the original CAD shape, proposing a slightly better design inspired by the sensitivities. Mathematically speaking the adjoint simulation provides the gradient of the chosen objective at the current baseline. Hence, the search direction in which to find an improvement is known while the step size for the variation is lacking. Heuristically, small moves are made and the process is repeated a few times (Christakopoulos and Müller, 2012). At the end, CFD simulations are again performed to check the achievements.

There are hybrid solutions, too, like parametric-adjoint techniques, see for instance (Robinson et al. 2012). The adjoint sensitivities and the parametric velocities (i.e., the changes of the shape when slightly perturbing parameters) are combined to yield improved shapes without deteriorating the quality of the surface. Fig.5 depicts an application within CAESES®. Here a fully-parametric model of the sports car's rear wing was combined with the viscous flow code iconCFD by ICON (www.iconcfd.com) as elaborated in (Brenner et al., 2014). The aim was to improve downforce without increasing drag while maintaining the wing's styling.

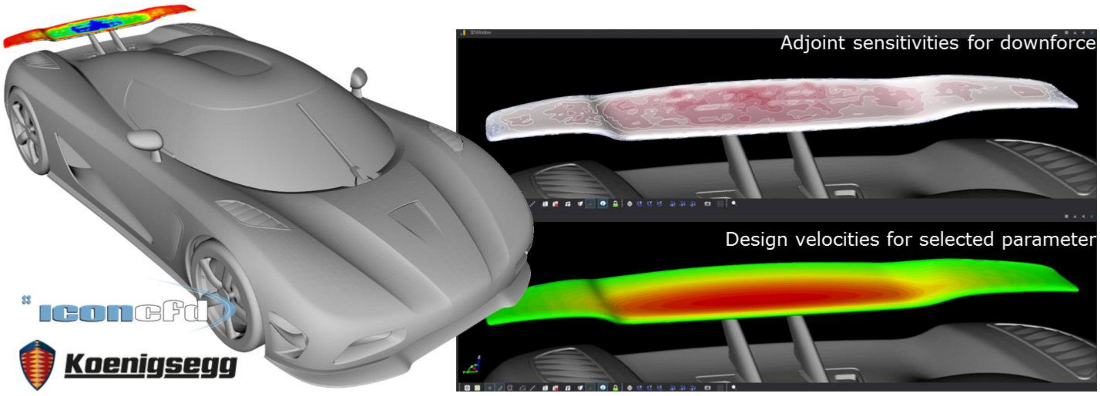

Fig. 5: Adjoint sensitivities for increase of downforce and design velocity from parametric model [By courtesy of Koenigsegg Automotive AB and ICON]

3.3 COMPARISON OF APPROACHES

The various parameter-based and parameter-free optimization approaches have their individual advantages – and weaker points. When selecting an approach, naturally, a design team's experience and the available engineering environment play important roles. Suggestions which approach to favor for which task are given in Tab.I.

Tab. I: Overview of shape optimization approaches

Approach		**Strong points**	**Weak points**	**Design stage**	**State**
Parameter-free optimization	**Topology optimization**	Innovative and unconventional designs Fast (one extended solver run)	Extra work in remodeling results in CAD tool Sometimes difficult to correlate with constraints, e.g. for production Limited to pre-defined objectives (e.g. pressure drop, homogeneity) Used only for internal flows Difficult to apply in multi-objective and multi-disciplinary scenarios	Concept design	Successful projects in selected industries Ongoing R&D
	Adjoint simulation	Fast (one additional solver run comparable to original simulation effort)	Confined to small changes (unless utilized repetitively) Limited to implemented objectives (e.g. drag, homogeneity) Difficult to apply in multi-disciplinary scenarios	Fine-tuning	Scientific applications Ongoing R&D, e.g. hybrid approach (parametric-adjoint)
Parameter-based optimization	**Fully-parametric modeling**	CAD geometry of high quality Allows for both global and local changes Direct incorporation of many constraints Applicable to multi-objective and multi-disciplinary optimization Creates overview and insight	Many solver runs Investment into parametric model needed By definition confined to design space of parametric model	Initial design to fine-tuning	Mature Successful projects in many industries Ongoing R&D, e.g. robust design optimization, HPC/Cloud
	Partially-parametric modeling	Quick and easy to set up and conduct Focus on local changes Applicable to multi-objective (and multi-disciplinary) optimization	Many solver runs By definition limited by design space of parametric model and baseline	Initial design to fine-tuning	Mature Widely used in many industries Ongoing R&D, e.g. usability

Since parameter-based optimization is quite mature and very popular the remainder of the paper shall be devoted to this approach.

The quality of a parametric model is decisive for the success of an optimization. This is because understanding an n-dimensional design space spanned by n free variables of the model – namely the parameters that shall be consciously changed – is anything but trivial. As a rule-of-thumb a design team needs to study about n times n variants to gain a reasonable appreciation of system behavior. (A statistically sound estimate involves many more factors (Siebertz et al., 2010).) If there are only two free variables, four variants would give a first insight. From a mathematical point of view it would allow a bilinear approximation of system behavior. If 10 free variables are involved, 100 variants ought to be evaluated. Working with many parameters or with parameters that are not really decisive quickly scales up the optimization task beyond all practical resources. Consequently, industrial projects rarely involve more than 20 to 30 free variables but rather show 10 to 15, possibly after the successive removal of less important parameters.

In order to keep the number of free variables as low as possible from the start a parametric model needs to be developed that suits the design and optimization task. An ideal parametric model (engineering model) for shape optimization is characterized as follows:

- All parameters are independent from each other,
- All potential variants are intrinsically fair (i.e., free of any unwanted shape characteristics),
- Many (geometric) constraints are readily incorporated,
- All variants are geometrically fit for simulation (e.g. free of gaps, folds, overlaps etc.),
- Shapes can be produced beyond the current engineering practice while avoiding unacceptable artifacts (well-balanced model).

The last requirement is the most challenging to satisfy: By definition all parametric models confine the potential outcome. In other words: Design freedom is deliberately reduced. Nevertheless, the model must allow for new shapes and should even contain some element of surprise. Otherwise nothing new can be found.

4. OPTIMIZATION PROCESS

Optimization is sometimes perceived as a black box that, magically, pops out the best design. Unfortunately, this is not quite right. Rather, apart from the automated number crunching, design studies and formal optimization are interactive processes with adjustments and reconsiderations, iteratively leading to design improvements and innovation.

4.1 TYPICAL PRACTICE

Parameter-based optimization processes – though they might differ in scope and conduct depending on the teams, tools and tasks at hand – have a common set of elements and a typical course of action: At the very start the team needs to discuss and agree on objectives, free variables and constraints. This is a critical part since all stakeholders need to express their expectations, bring in their knowledge about system boundaries and commit to the content of the project. Most projects then continue with a preparation phase during which a reasonable simulation set-up is established. Ideally, a grid variation study is undertaken to identify a resolution fine enough for acceptable accuracy, at least with regard to the correct ranking of variants, yet coarse enough for short turn-around time. Two distinct phases often follow: Exploration and exploitation. During the exploration phase the design space is scanned with the aim of identifying promising regions and understanding sensitivities, distinguishing parameters of higher importance from those with less influence. Sometimes the exploration results are used to build meta-models, i.e., response surfaces to replace the costly CFD in a subsequent step of exploitation. Next, an exploitation phase follows to squeeze out the best possible results. Commonly this is done within reduced regions, searching for local optima, or within a subspace in which some of the less important free variables are frozen. Finally, selected variants are analyzed and compared to the baseline, possibly at higher grid resolution and for conditions (and in disciplines) not considered during the optimization. From those the most favored variant is selected, concluding the project.

4.2 FORMAT

Optimization problems are cast into a standard format so that available mathematical techniques can be put to use without need for individual adaptations. Five elements have to be written down:

- Objective(s): What shall be improved (i.e., minimized or maximized)?
- Free variables: What can be changed (and is under control of the team)?
- Constraints: What needs to be observed (making a design feasible or infeasible)?
- Fixed parameters: What influences the system but is kept (or assumed) constant?
- Noise: What influences the system but is beyond control (e.g. scatter in material property)?

In shape optimization using CFD objectives and constraints are non-linear functions of the free variables. Free variables are mostly real numbers with some integers that then often represent topological information, e.g. number of blades. Free variables usually have lower and upper bounds which need to be chosen with care. Tight bounds give a small design space while loose bounds offer more room for improvements. If unsure one may commence with tighter bounds that are subsequently increased or shifted as the project matures.

Constraints are subdivided into inequality and equality constraints. Inequality constraints describe limits up to which a design is still acceptable. For example the shortest distance between the shape to optimize and another object must be larger than a given value (a hard-point constraint). Inequality constraints are often considered via penalty functions (or barriers). The idea is that objectives are artificially worsened by adding an extra term (i.e., the penalty) as soon as a variant is found to be infeasible. The penalty gets larger with the distance from the feasible domain (Gill et al., 1982). Equality constraints describe characteristics that need to be met exactly, e.g. the volume enclosed by a component has to equal a given value, cp. Fig.3. An elegant solution of handling an equality constraint is to incorporate it directly into the parametric model. This not only guarantees feasibility with regard to this constraint but may even help to reduce the number of free variables. As an alternative, equality constraints are relaxed to resemble inequality constraints, e.g. the volume has to be equal to or larger than the desired value. Occasionally, an equality constraint is combined with the objective via a so-called Lagrange multiplier, leading towards an extended objective at the cost of an additional unknown (Birk and Harries, 2003).

Sometimes it is quite obvious what shall be regarded as an objective (e.g. resistance, pressure drop) and what needs to be handled as constraints (e.g. distance to a hard-point). However, there are design tasks for which several performance measures can serve either as objectives or as constraints. If in doubt a final decision can be made on the basis of an exploration.

Taking the time to write down objectives, free variables and constraints explicitly is more important than it may initially seem. It is a team-building exercise to reflect what is really important and what may be omitted after all. Furthermore, it helps to avoid that further requirements come into play at a later point of time. For instance, if a new inequality constraint is introduced well into the project, many good designs may become infeasible all of a sudden (not a nice thing after days of number crunching).

4.3 COMPONENTS

Typically, a parameter-based optimization project involves the following components:

1. Variable geometry:
   A parametric model is developed and a shape variant is created as an instance of the chosen parameter values.
2. Pre-processing:
   The variant is pre-processed (e.g. generation of a watertight triangulation of the shape) to enable the simulation(s).
3. Simulation:
   For variants of interest one or several simulations are undertaken, by
   - Discretizing the fluid domain (e.g. by generating a volume mesh) and
   - Solving the governing flow equations.
4. Post-processing:
   Variants and their flow data are post-processed (e.g. visualizing flow fields for comparison) and, finally,
5. Optimization & Assessment:
   Variants are produced and assessed in accordance to the selected optimization strategy, repeating the sequence from variable geometry to post-processing again and again.

These five components constitute a synthesis model, tightly coupling CAD and CFD (see also Appendix Fig.A-1 and Fig.A-2). The actual sequence of creating a specific shape, discretizing this shape and, subsequently, the fluid domain, doing the number crunching, collecting the data and, finally, assessing the current design for objectives and constraints is repeated many times. The chosen algorithms, as will be explained below, decide how variants are brought about and how many are considered. Today, if simulations take a couple of hours per case several hundred variants are studied. Frequently, this is done over the course of a long weekend and, possibly, by distributing the heavy work load of simulation to an internal cluster or to a High Performance Computing (HPC) facility. If simulations can be completed quickly enough there are projects that even cover as many as several thousands of variants.

4.4 EXPLORATION

When coping with multi-dimensional design spaces a team has to trade resources against insight. To do so with high efficacy, Design-of-Experiments (DoE) have been developed. These are mathematical algorithms that create as much understanding as possible with as little cost as needed, see (Siebertz et al., 2010).

In general, an exploration – also called a design study – helps to

- Understand the design space and identify regions of interest,
- Find favorable variants, hopefully giving some performance improvements already,
- Evaluate sensitivities, possibly leading to a reduction of the number of free variables,

- Get an appreciation of trends and the potential for further optimization,
- Elucidate (if needed) what should be treated as an objective and what as a constraint,
- Provide the input data for building a meta-model if wanted.

The blunt approach for exploration is that of an exhaustive search in which a prescribed number of variants are created by equidistantly changing one parameter at a time. If you have two free variables and want to afford just three guesses in each direction you end up with a total of nine variants (i.e., a grid of three by three). For three variables you already need 27 variants (three by nine). It is easy to see that this scales up too quickly to be successful for expensive simulations. Consequently, other strategies were developed that deliberately leave out points in the design space that may not be absolutely necessary. Two popular algorithms are the Latin Hypercube (a factorial method) and the Sobol sequence. For an appreciation of the Sobol see Tab.II.

Tab. II: Sobol sequence for design studies (exploration)

The Sobol is a so-called quasi-random search which means that it produces a pattern in design space that looks random even though it actually is deterministic. The Sobol sequence fills up the design space (namely a hyperspace of *n* dimensions) such that the next variant is always placed in the region that is least populated so far (Press et al., 2007). Its behavior somewhat mimics people at a beach. Consider the beach to be a two-dimensional search space. The first person to arrive will probably place his or her towel somewhere close to the center. The next person will choose a spot far away from the first one (unless there is some non-mathematical attraction that we shall disregard). The third person will do likewise, now trying to lie down at a comfortable distance to both the first and the second person. This carries on until the beach is somewhat covered, Fig.II-1. (Similar observations can be made in elevators where strangers naturally find an unbiased distance towards each other, even dynamically by reshuffling every time someone comes in or goes out.)

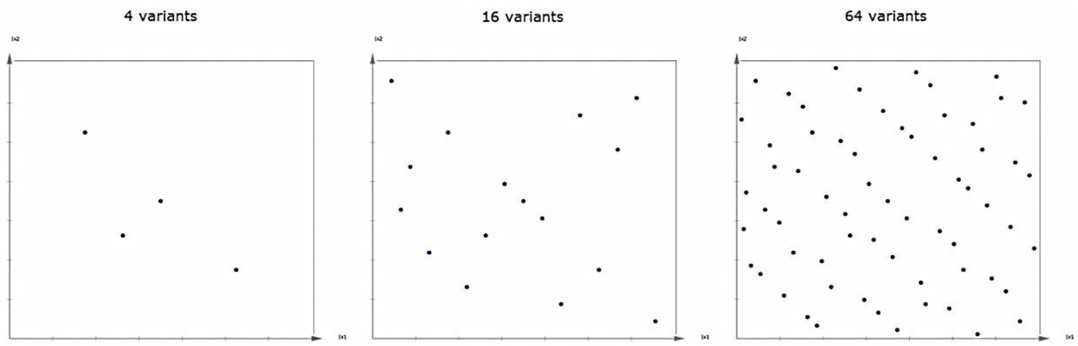

Fig. II-1: Different number of variants as distributed by Sobol in two-dimensional space

However, unlike people at the beach the Sobol is a mathematical strategy that comes without any true randomness. This is very convenient as it easily permits repeating a specific search. For instance, after running an expensive study you realize that some outputs of interest are missing. It is then straightforward to repeat the Sobol, getting exactly the same parameter values as before, only for the additional analyses. Another strong suit is that large investigations can be subdivided into more manageable portions. For instance, you run the first 50 variants (0 to 49) on one workstation and the next 50 variants (50 to 99) on another computer. Both patterns combined behave as one large search, meaning exactly the same as if you had done all variants in one single sweep (0 to 99). Naturally, this behavior can also be utilized for a successive extension of the exploration phase. If you feel you need more information after 100 variants have been completed you can just carry on with the next 100 without any overlap. Moreover, since every variant is independent of any other the entire DoE can be executed in parallel. An attractive side effect of this independence is that certain constraints can be checked for each variant before actually triggering any expensive CFD runs (e.g. threshold on grid quality), saving the resources for simulations where they are most valuable. In addition, it does not really hurt so much if the evaluation of one or several variants fails (e.g. due to a meshing problem or a diverging flow solution) as this would not terminate the process.

4.5 EXPLOITATION

As soon as regions of interest in the design space are known from the exploration phase an exploitation phase – also called formal optimization – is started. Its purpose is to find further designs that outperform all designs known so far.

Usually, several optimization runs are conducted. The team selects a handful of the more promising designs from the exploration phase (or simply considers a few manually created starting points). Optimization strategies are then put to use to systematically change the free variables in order to iteratively advance towards (at least local) optima.

Ideally, the exploitation would yield a true (and even a global) optimum. However, resources rarely permit that optimality conditions are strictly met. For mathematical proof first and second derivatives of the objective(s) with respect to all free variables would have to be computed. This is very expensive as it calls for numerical approximations, requiring a lot of additional CFD analyses. As a consequence, a more humble approach is taken in most industry projects. The best variant is simply chosen from several improved designs. Any improvement of the objectives is welcome and happily lived with ever after. Pragmatically, even though nobody actually knows if some superior design was still out there, yet to be discovered, the final shape is at least better than its baseline.

Many different formal optimization strategies are available, see e.g. (Birk and Harries, 2003) for an overview. At a high level they are categorized into local (mostly deterministic) and global (primarily stochastic) methods. Local methods are often further subdivided into gradient-free and gradient-based methods such as the Simplex algorithm (Nelder and Mead, 1965) and the conjugate-gradient method (Hestenes and Stiefel, 1952), respectively. So as to understand the principle of formal optimization the Simplex algorithm, being a very popular strategy, is explained in Tab.III.

Tab. III: Simplex for formal optimization (exploitation)

The Simplex is an easy yet representative deterministic strategy. In *n*-dimensional space a Simplex is an object of *n+1* corner points (a triangle in 2d, a tetrahedron in 3d etc.). Let us again take a beach for illustration. It is a two-dimensional domain and, hence, we would have two plus one corner points to work with. If the objective was to find the highest dune (giving the best view) the Simplex goes about as follows: Three people are placed reasonably close to each other, say 5m apart on a beach of 100m by 100m. They hold three flexible ropes, forming a triangle. For our local search the person who is at the lowest position with respect to sea level has to change position while the other two persons stay where they are. The person to move, naturally hoping to improve altitude, now walks towards the rope connecting the two stationary persons, crosses the rope in the middle and carries on to the other side until a new triangle of similar size has been established (so-called reflection). The objective, here the altitude, has now to be determined for the new position. In principle, the search is then repeated, namely the person that now happens to be at lowest altitude needs to move. This is done again and again until the highest point is found (or the maximum number of iterations has been reached). There are several situations which require special treatment, namely expansion (to realize faster gains), contraction (to counteract overshooting) and compression (to reduce the size of the Simplex and, eventually, to terminate the search). The initial Simplex is typically derived from the exploration phase. For the beach we would ask all visitors, e.g. quasi-randomly distributed according to the Sobol (Tab.II), what their individual altitudes are. Heuristically, the person that is at the highest position so far makes a sensible starting point. The *n* closest neighbors may complement the Simplex, provided their distances relative to the size of the design space and with respect to the behavior of the objective are reasonable (not too small and not too large). Alternatively, little side-steps (say 5m at the beach) can be made by changing one free variable at a time (i.e., going parallel to the coordinate axes). This, however, requires *n* additional evaluations of the objective. Intuitively, it makes sense to undertake a handful of local searches in different areas wherever further gains can be expected from good candidates found during the exploration.

Global methods subsume many quite diverse strategies of which particle swarm optimization, simulated annealing and, notably, genetic algorithms (Goldberg, 1989) are very popular. In a way, global methods combine the two phases of exploration and exploitation. As they sweep over the design space they generate both an overview and design improvements, albeit at the cost of very high numbers of functional evaluations. Due to their need for many hundreds of CFD runs, even for problems with small sets of free variables, global methods are only utilized by teams with exceptional resources.

4.6 COMPARISON OF ALGORITHMS

There is no optimal optimization method. Rather, the choice depends on the design task at hand as well as the time and computer power available. Furthermore, the comfort of having worked successfully with one method might just lead to applying it again for the next project. Table IV gives an overview of representative algorithms, helping to make a selection.

Tab. IV: Overview of optimization algorithms

Phase	Purpose	Prominent algorithms	Strong points	Weak points
Exploration (Design study)	Understand design space Find good starting points for exploitation Get relationships between objective(s) and free variables (sensitivities) Get appreciation of optimization potential Help decide on objectives and constraints (if unclear) Provide data for meta-models (if pursued)	Sobol Latin Hypercube Taguchi method Exhaustive search (not a true DoE)	Variants are independent of each other and can be evaluated in parallel (robust execution) Derivative free Easy to comprehend (good feeling)	Intrinsically resource intensive for large design spaces with many free variables
Exploitation (Formal optimization)	Find local optima	Simplex T-Search	Gradient-free (robust execution)	May overshoot and end up in different local region than started from Each design has to be successfully evaluated (sequential execution)
		Conjugate-gradient method	Fast for well-behaved problems (e.g. nearly quadratic behavior)	Gradients need to be approximated numerically, e.g. by means of forward differencing (with challenge to identify appropriate step size) Needs line search (or similar) to advance towards local optimum
Combined exploration and exploitation	Understand design space and identify optima Determine Pareto front	MOGA (multi-objective genetic algorithm) MOSA (multi-objective simulated annealing) Particle swarm optimization	Variants (of one generation or time step) are independent of each other and can be evaluated in parallel Robust execution (failure of one or several designs does not lead to termination) Generates good insight Easy to apply to multi-objective and multi-disciplinary design tasks	Very expensive to conduct

N.B. Many excellent optimization algorithms have been proposed and the table does not aim at completeness. References for further reading are given in section 6.

Generic optimization environments provide a range of tools to choose from. Popular algorithms for practical shape optimization using CFD are the Sobol sequence for exploration and the Tangent Search Method (T-Search) for exploitation as offered within CAESES®. The T-Search was originally proposed by (Hilleary, 1966). It combines smaller steps and larger moves through the design space (a pattern search) and directly handles inequality constraints, see (Birk and Harries, 2003) for an elaboration. Mathematically speaking it is a gradient-free method but it comes up with probing moves not dissimilar to gradient directions. The examples shown below are primarily based on employing the Sobol and the T-Search.

5. EXAMPLES

Several examples are presented to illustrate practical shape optimization using CFD. The examples stem from industry projects realized by coupling CAESES® to the state-of-the art RANS solvers utilized by the different design teams on a daily basis.

Further examples along with animations can be found at www.caeses.com/industries/case-studies.

5.1 DIFFUSOR DESIGN

For a diesel engine the hot exhaust gas had to be diffused and, in addition, turned by 90° due to constraints in space. The component is relatively simple but occurs repeatedly and, hence, influences the overall performance of the system. STAR-CCM+ by CD-adapco (www.cd-adapco.com) was applied to compute pressure loss and flow homogeneity. Both were treated as objectives and considered at the diffusor's outlet plane and further downstream where the hot gas reached the next sub-system.

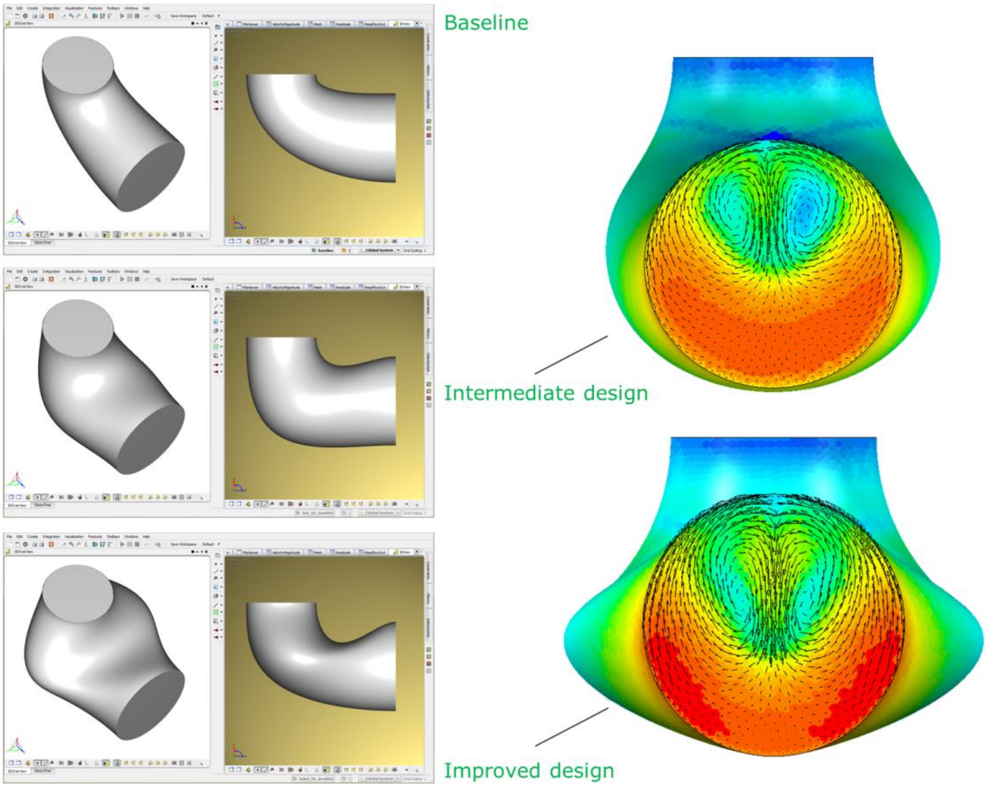

Fig. 6: Improvement of a 90° diffusor for a diesel engine

A DoE was conducted on the basis of a fully-parametric model that captured all constraints and allowed capturing the volume distribution along the connecting path between the diffusor's inlet and outlet. Fig.6 depicts the baseline, an intermediate design and the improved design along with flow visualizations at the outflow plane for two of the better shapes. The baseline with its simple sweep of circular sections yielded a pressure loss of 47 mbar (flow rate of 1 kg/s at 500°C) while axial flow homogeneity was 0.51 (1 being the ideal). In comparison, the final design with its "Cobra-style" shape, Fig.6, resulted in a substantially reduced pressure loss of only 14 mbar and an increase to 0.87 in homogeneity.

5.2 FUNNEL DESIGN

For a mega-yacht the exhaust funnel was studied with regard to gas contamination on the upper deck while at anchor in calm weather, representing the so-called "party condition." The fluid domain was discretized using ICEM while the simulations of the external thermo-fluid dynamic field were performed with ANSYS CFX (www.ansys.com). For details see (Harries and Vesting, 2010).

Fig.7 shows an impression of the yacht along with its plume as emitted from the auxiliary engines, close-ups of the grid and two representative designs. The parametric model within CAESES® allowed varying the length and angle of the exhaust pipes along with the size and shape of various deflectors. As an objective for minimization the volume fraction of exhaust gas integrated over a plane downstream of the funnel was considered. An overnight Sobol of 50 variants was run by CAESES® on a small cluster so as to identify unfavorable and favorable designs as needed for styling decisions. It can be seen in Fig.7 that longer and steeper funnel pipes in connection with extended and more strongly curved deflectors yield tangible reductions of exhaust gas (the best design being about 40% better than the worst).

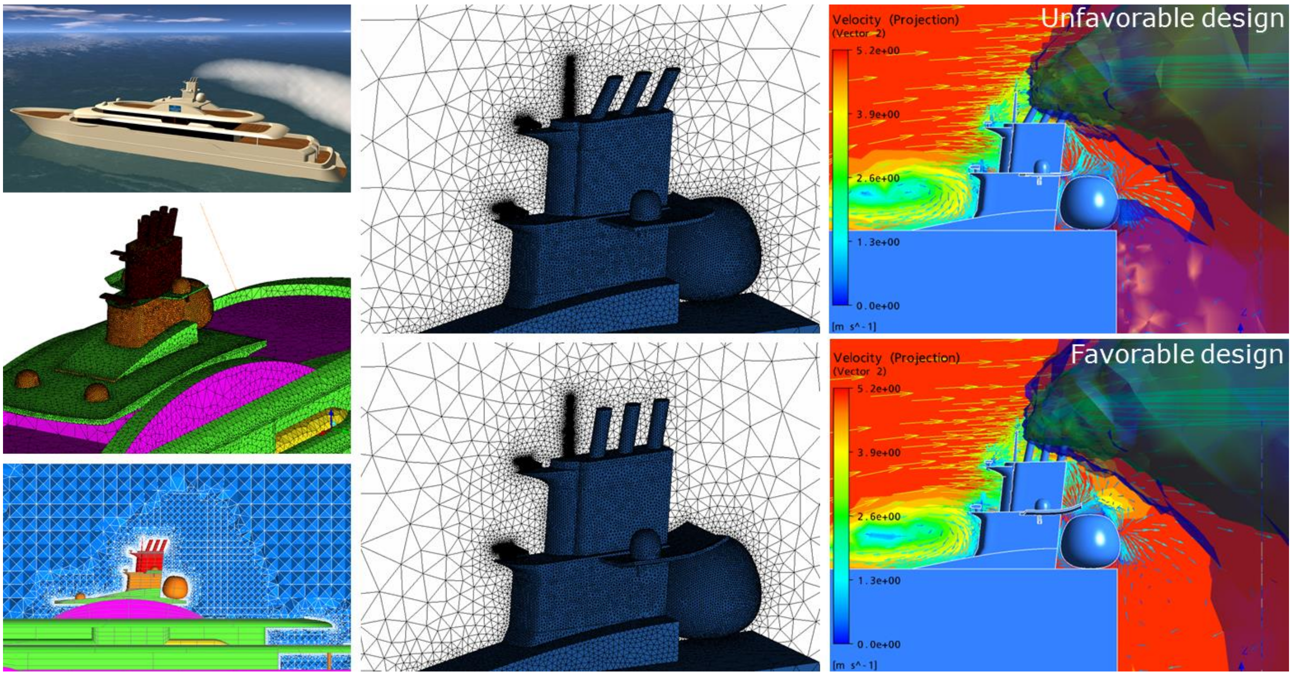

Fig.7: Funnel design study with unfavorable and favorable designs from DoE
[Compiled from (Harries and Vesting, 2010)]

5.3 VALVE DESIGN

A control valve as used in a wide range of industries was optimized by coupling CAESES® to Simulation CFD by Autodesk (www.autodesk.com). While the cylindrical plug was kept unchanged the valve housing was varied with the objective to reduce pressure drop. A robust and flexible engineering model was created in CAESES® to control the free-form shape of the housing, deviating from the classical sphere. The model was fully-parametric with parameters that would control the dividing contour, the housing's transition to the pipes and its curvature distribution at the apex.

An exploration was run, using a Sobol, which was followed by an exploitation, using a T-Search. The best design yielded a 5.6% reduction in pressure drop. After careful considerations, the bounds of several free variables were relaxed and an additional T-Search was run, eventually leading to 27% less pressure loss for the optimized design over the baseline. Fig.8 depicts the baseline, an intermediate design (from the first T-Search) and the optimized design along with their corresponding flow fields.

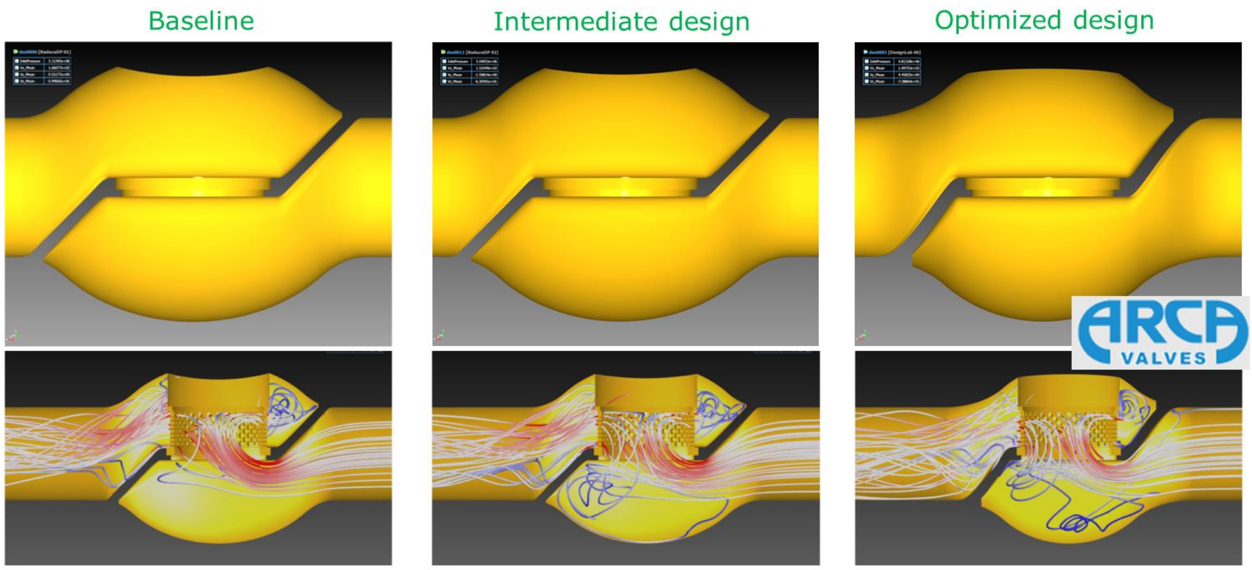

Fig.8: Evolution from baseline, intermediate design towards optimized design
[By courtesy of ARCA (www.arca.de)]

5.4 RADIAL COMPRESSOR DESIGN

The volute and diffusor of a radial compressor were optimized for isentropic efficiency (Ratz, 2014). CAESES® was coupled to software systems by Numeca Int. (www.numeca.com), specifically, Autogrid5 and Hexpress for compound grid generation and FINE/Open for flow simulation. The geometry was described in CAESES® by means of a fully-parametric model. The volute's cross section was defined by several parameters (such as the radius as a function of peripheral angle) which themselves were subjected to change in circumferential direction, cp. Fig.2. This enabled shape adaptations for given mass flow and allowed controlling the volute's diffusion characteristics. The parametric model of the diffusor comprised the stagger angle, the blades' twist and their pitch and trailing edge positions. These parameters were changed circumferentially, too, allowing the diffusor to be non-periodic (according to a user-specified "normal" distribution). The impeller was left unchanged but accounted for in the simulation.

During the optimization CAESES® was run in batch-mode to provide the variable geometry while Numeca’s Design3D was chosen to control the process, using a genetic algorithm in combination with an artificial neural network as a meta-model.

Fig.9 illustrates the project. Comparing the baseline to the best design a small rotational shift (by 4.2°) was observed between the original diffusor blades and the newly identified optimal blade configuration. Further to this, the stagger angle was increased (by 1.4°) and the position of the trailing edge was moved in radial direction (by 1.5 mm). The volute of the optimal design featured a slightly larger area distribution. Isentropic (peak) efficiency was increased by 1% (at 5.9 kg/s) with performance improvements over the entire operating range.

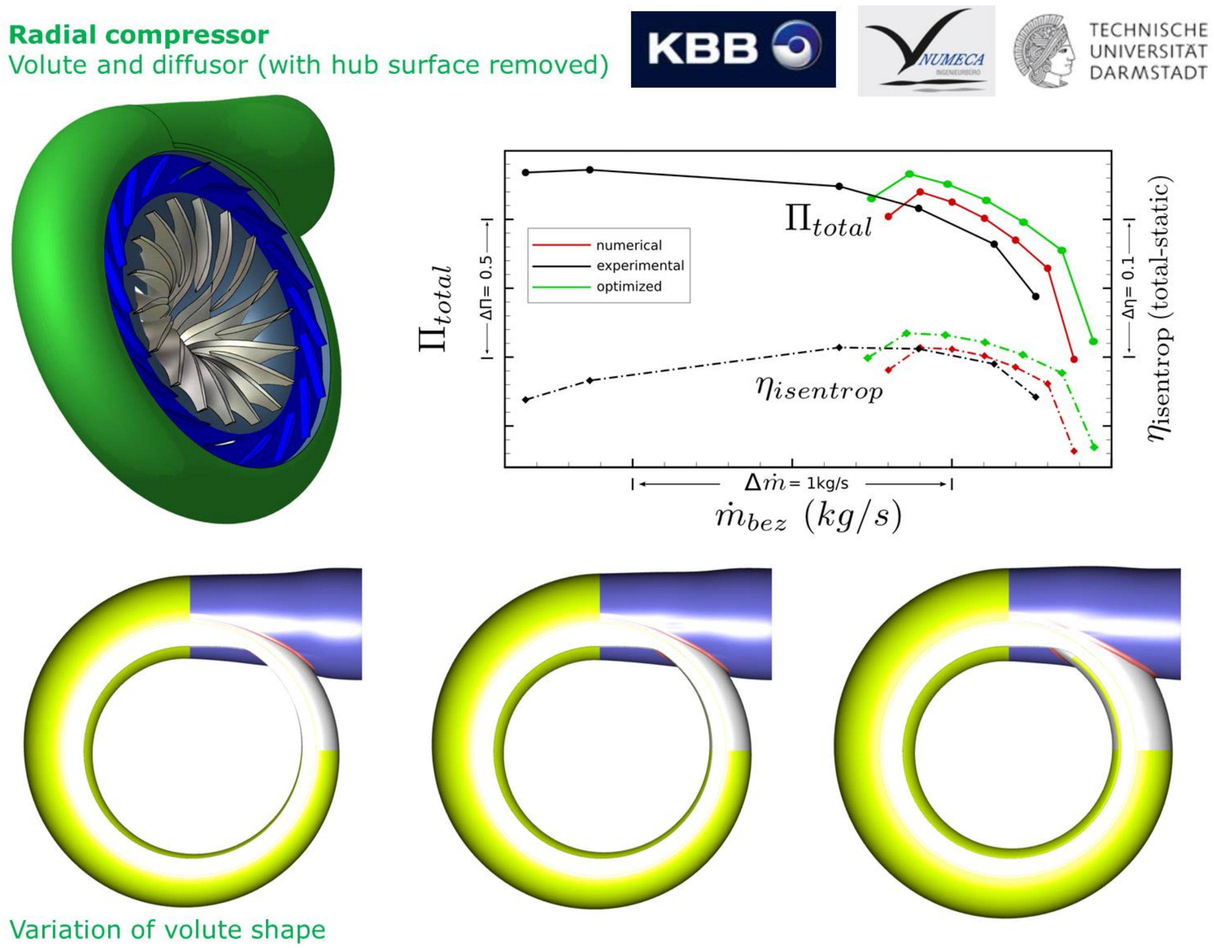

Fig.9: Optimization of a radial compressor by means of parametric variations of volute (green) and diffusor (blue) [By courtesy of Technical University Darmstadt (www.tu-darmstadt.de), Numeca Ingenieurbüro (www.numeca.de) and KBB (www.kbb-turbo.de)]

5.5 PROPULSION SYSTEM DESIGN

A propulsion system was developed by Voith for the dynamic positioning of large offshore structures on the basis of CAESES®. The system comprises a ducted propeller, a nozzle and a strut, see Fig.10. Starting from a manually designed baseline (red circle in Fig.10) many variants were investigated at a time, finally resulting in several thousand variants analyzed with RANS simulations.

The parametric model for the propeller blade contained descriptors for camber, chord length, pitch, rake, skew and thickness while the duct was defined via proprietary profiles. The shape and inclination of the nozzle were parameterized and varied, too. As is common in maritime propulsion, thrust and cavitation volume are two important objectives, bringing about a multi-objective problem (as is further explained in section 6). It can be nicely seen in Fig.10 that designs could be found that yielded higher thrust for the same cavitation volume (e.g. green circle). Vice versa, designs were realized that gave the same thrust for lower cavitation volumes (e.g. blue circle). The quadrant left and above of the baseline offers designs that are all better with respect to both objectives (creating a Pareto front as depicted in light-blue).

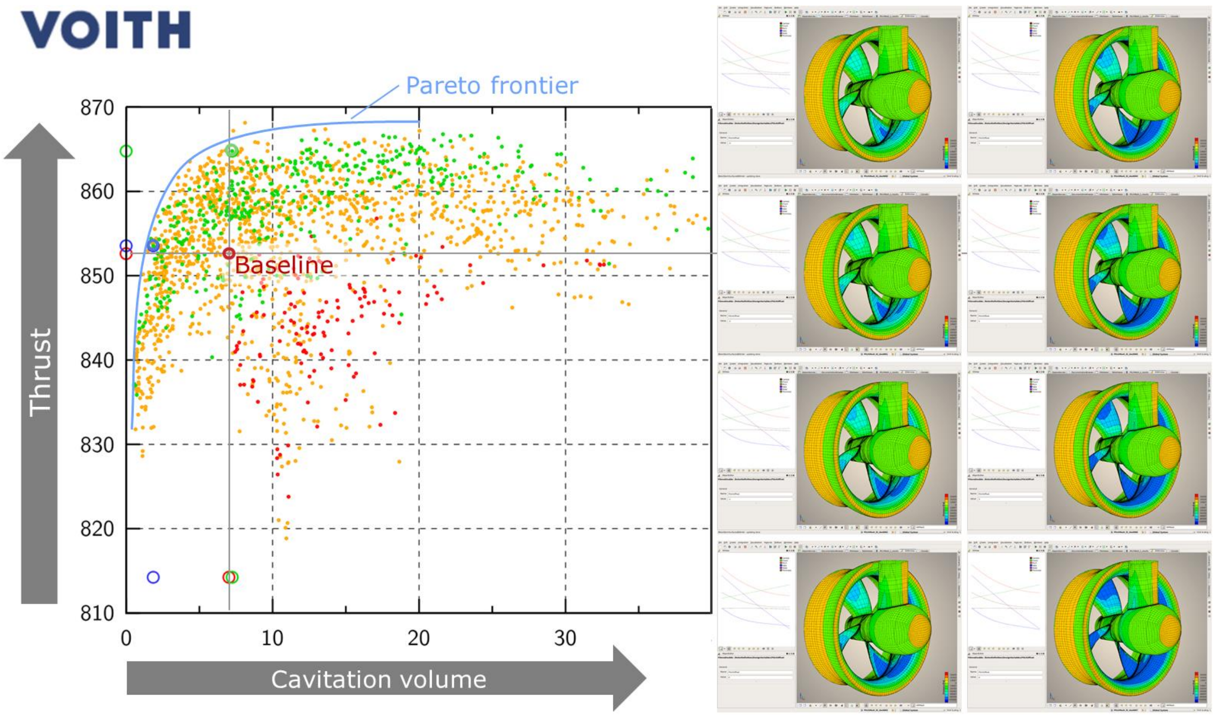

Fig.10: Comparison of performance for competing objectives for hundreds of variants along with representative instances of the propulsion system [By courtesy of Voith Turbo (www.voith.com)]

6. FURTHER ISSUES

As pointed out at the beginning of the paper, many important issues are not elaborated here for brevity. Nevertheless, they should be mentioned in order to broaden the picture.

To start with, the world is multi-objective and so is design. Sometimes it is possible to express all objectives in the same unit (mostly in economic terms) or to normalize and mix objectives that are of different nature. Very frequently, however, objectives cannot be made directly comparable. To solve this predicament Pareto fronts can be studied, Fig.10. If a feasible design cannot be further improved for one objective without deteriorating any other objective, the solution belongs to the Pareto set. All these so-called non-dominated designs are respectable candidates for selection and it depends on the preference of the decision makers which one is considered to be best for the design task at hand. (To this end utility functions may serve to rank variants.)

Multi-objective optimization often refers to looking at several objectives that belong to the same engineering discipline, say hydrodynamics. In naval architecture a typical multi-objective problem is to consider resistance, propulsion and sea-keeping at the same time. A multi-disciplinary optimization is established as soon as several disciplines are involved, say hydrodynamics, structures and economics. An engineering example of multi-disciplinary optimization is presented in (Harries et al., 2011). A tanker was optimized for objectives such as fuel consumption, strength, oil outflow in case of accidents (OOI as a measure for safety), required freight rate (RFR as a combined economic target) and energy efficiency design index (EEDI as a regulatory requirement). The design synthesis model called for the combination of quite a few tools, each covering a specific field of engineering and being controlled by CAESES®. Fig.11 gives the results aggregated from a DoE with RFR vs. deadweight (i.e. a ship's carrying capacity). It can be observed that there is no single answer but that different designs perform best for different preferences.

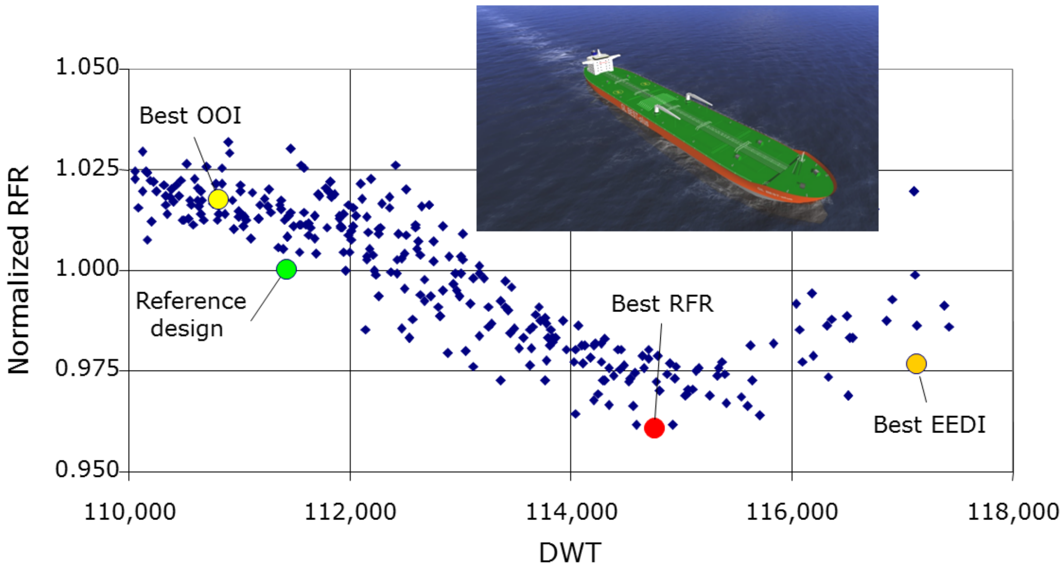

Fig. 11: Results of a multi-disciplinary optimization project
[Reproduced from (Harries et al., 2011)]

An additional issue of any optimization is the sensitivity of a design with regard to smaller (e.g. noise) and larger differences (e.g. operational profile) between the idealized situation and the real world. There exists a natural contradiction between optimality and robustness. A solution may be exceptionally good for one objective in one given situation while performance quickly drops if conditions change just a bit. Clearly, this is something to avoid. Robust optimization is an active field of research and development, addressing this predicament. See (Beyer and Sendhoff, 2007) for a through treatment. An ad-hoc solution is to formulate a range of conditions for which the design shall perform really well, essentially creating a multi-objective problem.

Sometimes, this is again cast into a single-objective problem in which several objectives are normalized (e.g. with baseline performance), weighted (e.g. according to frequency distribution) and summed up. For example, a ship typically sails at different speeds and drafts. A handful of loading conditions are combined so as to yield the objective for a robust optimal solution. Similarly, the blades of a turbine have to be really good at design point but also need to deliver a large operating range. Again, so as to avoid overbreeding a few representative conditions can be taken into account simultaneously.

Sensitivities also need to be considered in the context of noise and deterioration. The former is typically associated with fluctuating conditions such as spread in material property and deviations in production accuracy. The latter may simply stem from the wear and tear encountered after years of work. Therefore, it needs to be judged if the ranking of variants will actually remain true even if conditions slightly differ or change. Let us take again an example from the maritime industry. In practical hull form optimization naval architects neglect hull fouling. It is assumed that hull resistance will increase by about the same amount for all variants. As a consequence, the hull shape that yields best performance when hydrodynamically smooth should still be the best hull after surface roughness has come into play since all variants are equally affected.

More issues and questions, e.g. low-fidelity vs. high-fidelity simulations, are posed in (Harries, 2008). Not surprisingly, there are numerous papers and many books on the subject of optimization, see e.g. (Eschenauer et al., 1990), (Birk and Harries, 2003), (Onwubolu and Babu, 2010) and the websites given in the reference list. These sources may serve as entry points for further reading.

7. SUMMARY

Different approaches of shape optimization using Computational Fluid Dynamics have been presented.

Parameter-based optimization is the most popular approach and is applied during both initial and detailed design. Increasingly, shape optimization using CFD is done early on – i.e., upfront – as the insight and the potential for performance gains are highest. In order to be able to undertake optimization projects special parametric models are required, focusing on engineering rather than production. They encompass partially-parametric (e.g. free-form deformation) and fully-parametric models. The former utilize existing shapes which are modified parametrically. The latter allow the creation and variation of shapes from scratch. Dedicated upfront CAD software for variable geometry complements the traditional CAD software, offering parametric modeling as needed for shape optimization.

Parameter-based optimizations frequently comprise exploration (design study) and exploitation (formal optimization). During the exploration phase a reasonable number of shape variants is created and analyzed with the main aim of understanding the design space, identifying favorable improvements and evaluating sensitivities. Design-of-Experiments are mathematical strategies to undertake explorations systematically and economically. During the exploitation phase the shape is modified methodically with the aim of further improving one or several objectives (performance measures) that are considered to be representative of the quality of the product. Quite a range of strategies are available. Which strategy best suits a particular design task depends on many things, for instance the simulation time needed, the available resources and the conditioning of objectives and constraints.

Several examples have been given to illustrate applications from different fields of engineering. Shape optimization using CFD is prominent in the automotive, aerospace and maritime industries as well as in turbomachinery and mechanical engineering. This is due to the fact that concerted shape changes can bring about decisive improvements for typical objectives such as resistance and lift, flow homogeneity and pressure drop. In general, energy efficiency is a major driving force for shape optimization.

ACKNOWLEDGEMENT

I would like to thank the team at FRIENDSHIP SYSTEMS, in particular key developers and application engineers of CAESES®: Arne Bergmann, Erik Bergmann, Mattia Brenner, Konrad Lorentz, Jörg Palluch, Park Daehwan, Stefan Wunderlich, Mike Saroch and Benjamin Zeitz. Furthermore, I would like to acknowledge the contributions by my long-term colleague Claus Abt, not only in writing this whitepaper but also in pushing shape optimization using CFD. Finally, I would like to thank the companies and their representatives that provided the examples picked for illustration.

REFERENCES

Abt, C.; Harries, S.: A New Approach to Integration of CAD and CFD for Naval Architects, 6th International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2007), Cortona, Italy, April 2007

Beyer, H.G.; Sendhoff, B.: Robust optimization – A comprehensive study, Computer Methods in Applied Mechanics and Engineering 196 (2007), 3190-3218, Elsevier 2007

Birk, L.; Harries, S. (Edt.): OPTIMISTIC – Optimization in Marine Design, Mensch & Buch Verlag, ISBN 3-89820-514-2, Berlin, Germany, May 2003

Brenner, M.; Zimmer, A.; Weickgenannt, S.: Effiziente Optimierung von strömungsbeeinflussten Geometrien durch Übertragung von adjungierten Sensitivitäten auf die CAD Modellparameter, NAFEMS Konferenz: Berechnung und Simulation – Anwendungen, Entwicklungen, Trends (in German), Bamberg, Germany, May 2014

Christakopoulos, F.; Müller, J.-D.: Discrete adjoint CFD solver for first and second derivatives via Automatic Differentiation, Conference on Industrial Design Optimisation for Fluid Flow, Munich, Germany, March 2012

Eschenauer, H.; Koski, J.; Osyczka, A. (Edt.): Multicriteria Design Optimization – Procedures and Applications, Springer-Verlag, 1990

Gill, P.E.; Murray, W.; Wright, M.H.: Practical Optimization, Academic Press, London, 1982

Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison Wesley, 1989

Harries, S.: Serious Play in Ship Design, Tradition and Future of Ship Design in Berlin, Colloquium, Technical University Berlin, Berlin, Germany, February 2008

Harries, S.: Investigating Multi-dimensional Design Spaces Using First Principle Methods, 7th Int. Conference on High-Performance Marine Vehicles (HIPER 2010), Melbourne, Florida, USA, October 2010

Harries, S.; Abt, C.; Hochkirch, K.: Modeling meets Simulation – Process Integration to improve Design, Honorary colloquium for Prof. Hagen, Prof. Schlüter and Prof. Thiel, Duisburg, Germany, July 2004

Harries, S.; Tillig, F.; Wilken, M.; Zaraphonitis, G.: An Integrated Approach for Simulation in the Early Ship Design of a Tanker, 10th International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2011), Berlin, Germany, May 2011

Harries, S.; Vesting, F.: Aerodynamic Optimization of Superstructures and Components, 9th International Conference on Computer Applications and Information Technology in the Maritime Industries (COMPIT 2010), Gubbio, Italy, April 2010

Hestenes, M.R.; Stiefel, E.: Methods of Conjugate Gradients for Solving Linear Systems, Journal of Research of the National Bureau of Standards 49 (6), 1952

Hilleary, R.: The Tangent Search Method of constrained minimization, Tech.Rep./Res.Paper 59, United States Naval Postgraduate School, Monterey, USA, 1966

Nelder, J.A.; Mead, R.: A Simplex method for function minimization, Computer Journal 7, 308–313, 1965

Onwubolu, G.C.; Babu, B.V.: New Optimization Techniques in Engineering, Springer Berlin Heidelberg, 2010

Press, W.H.; Teukolsky, S.A.; Vetterling, W.T.; Flannery, B.P.: Numerical Recipes: The Art of Scientific Computing, Cambridge University Press, 2007

Robinson, T.T.; Armstrong, C.G.; Chua, H.S.; Othmer, C.; Grahs, T.: Optimizing Parameterized CAD Geometries Using Sensitivities Based on Adjoint Functions, Computer-Aided Design & Applications, 9(3), 2012, 253-268, CAD Solutions, LLC

Ratz, J.: Simultaneous optimization of diffuser with volute for applications in radial compressors, master thesis (in German), Technical University Darmstadt, July 2014

Sederberg, T.W.; Parry, S.R.: Free-form deformation of solid geometric models, SIGGRAPH Computer Graphics (ACM) 20 (4): 151–160, 1986

Siebertz, K.; van Bebber, D.; Hochkirchen, T.: Statistische Versuchsplanung, Design of Experiments, Springer-Verlag Berlin Heidelberg, 2010

Stephan, M.; Häußler, P.; Schützle, T.; Johannsen, D.: An Efficient Approach for CFD Topology Optimization, NAFEMS, Wiesbaden, Germany 2008

Stück, A.; Kröger, J.; Rung, T.: Adjoint-based Hull Design for Wake Optimisation, Ship Technology Research, 58(1):34-44, 2011

http://www.caeses.com

http://www.cs.sandia.gov/opt/survey

http://www.mat.univie.ac.at/~neum/glopt/software_g.html

http://www.optimization-online.org

http://plato.asu.edu/guide.html

GLOSSARY

Baseline	Initial shape, parent design (starting point)
Bounds	Lower and upper limit for each free variable
CAD	Computer Aided Design; traditional CAD systems aim at a detailed models, including data for production
CAE	Computer Aided Engineering
CAESES	Computer Aided Engineering System Empowering Simulation, see www.caeses.com
CFD	Computational Fluid Dynamics
Constraint	Limit that separates feasible from infeasible solutions within the design space
Design space	*n*-dimensional space (hyperspace), *n* being the number of free variables in which variants differ
Design of Experiments (DoE)	Method to get to know the design space by gathering maximum information with as few variants as possible (popular methods are Sobol and Latin Hypercube)
Equality constraint	A characteristic of the design that has to be met exactly; often a non-linear function of the free variables
Exploitation	Formal optimization, controlled via a mathematical strategy (popular methods are deterministic searches such as Simplex and genetic algorithms such as MOGA)
Exploration	Design study, typically conducted by means of a Design of Experiments (DoE)
Free variable	Parameter under control of the design team and deliberately changed during an optimization
Fully-parametric modeling	CAD approach in which the entire geometry of a product is defined by parameters
Functional surface	Shape that determines fluid-dynamics performance of a product (often featuring compound curvature)
Inequality constraint	A characteristic of the design that needs to be less than (or equal to) or greater than (or equal to) a given value; often a non-linear function of the free variables
Multi-disciplinary optimization	Optimization that tries to improve objectives stemming from different fields of engineering (e.g. thermo-fluid dynamics, structures and economics)
Multi-objective optimization	Optimization that tries to improve two or more objectives at the same time, resulting in dominated and non-dominated variants (Pareto front); objectives are from the same field of application (e.g. hydrodynamics only)
n	Number of free variables, defining the design space
Objective	Objective function, target, goal, measure of merit
Pareto front	Also Pareto frontier, i.e., the set of (non-dominated) variants that cannot be further improved with regard to one objective without impairing any other
Partially-parametric modeling	CAD approach in which changes to an existing geometry are described by parameters

RANS code	CFD code solving the Reynolds-averaged Navier-Stokes equations (RANSE)
Response surface	Also meta-model and surrogate model, i.e., a model built on top of another model to approximate system behavior, typically replacing expensive simulations; Polynomial regression, Kriging and Artificial Neural Networks are common techniques
Single-objective optimization	Optimization that tries to improve one objective only (but the objective may be a weighted sum of several normalized objectives)
Simulation	Attempt to predict aspects of the behavior of a system by creating an approximate (mathematical) model of it while omitting certain (less important) characteristics
Simulation-driven design	Design approach utilizing numerical simulations at a large scale so as to develop a product; shape changes are brought about through simulations
Upfront CAD	Computer Aided Design for the conceptual and preliminary design phases (and for fine-tuning); upfront CAD systems allow to build engineering models for design studies and formal optimization
Variant	Instance of a model, shape, object or product investigated during an optimization

N.B. Many different words and terms for the same things are found in the optimization literature, depending on certain schools and on fields of application. This Glossary is meant primarily as a look-up table for the terminology used here.

ABOUT THE AUTHOR

Dr.-Ing. Stefan Harries *MSE* is member of the executive board of FRIENDSHIP SYSTEMS AG. He studied mechanical engineering at the Technical University Darmstadt, Germany, and naval architecture at both the Technical University Berlin, Germany, and the University of Michigan, USA. He received his PhD in Computer Aided Engineering in 1998. He co-founded FRIENDSHIP SYSTEMS in 2001 and has been associated with TU Berlin as external lecturer for Computer Aided Design of Marine Systems since 2004.

CONTACT DETAILS

Please get in touch. Questions, feedback and suggestions for improvement of the paper are highly welcome: whitepaper@friendship-systems.com.

The field of shape optimization is vast. If you have material that you feel should be considered in a reviewed version of this whitepaper, please do not hesitate to send it (preferably as a pdf-file or a download link).

APPENDIX

The examples shown in the paper were realized with CAESES® / FRIENDSHIP-Framework. If you want to use CAESES® simply download the latest version from www.friendship-systems.com (for both Windows and Linux).

CAESES® / FRIENDSHIP-Framework comprises variable geometry, pre-processing, software connection to external simulation codes, post-processing and optimization & assessment, see Fig. A1. The community edition CAESES® Free can be run free of charge, also within commercial projects.

Please note that CAESES® does not offer any CFD code but rather connects to any simulation code that can be run in batch mode, Fig. A2.

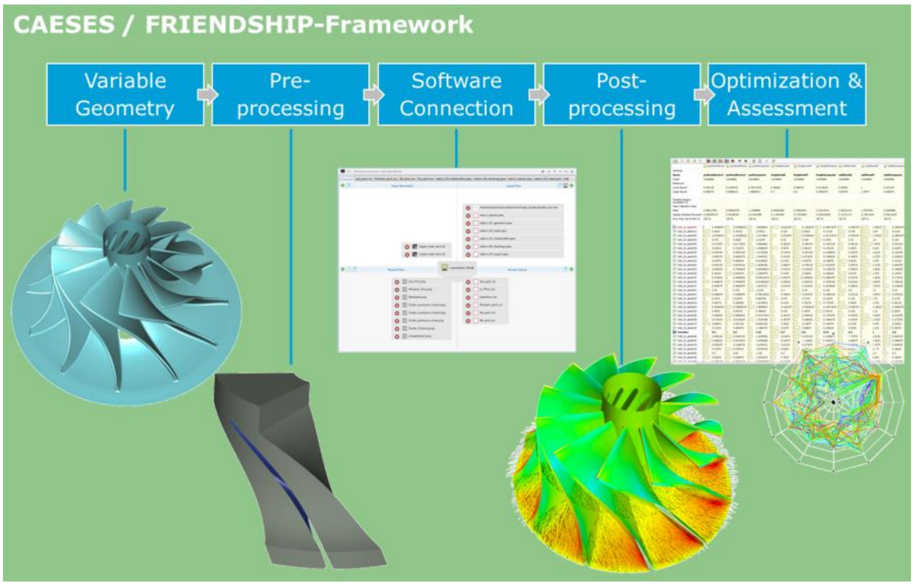

Fig. A-1: Illustration of components of an optimization process using CFD

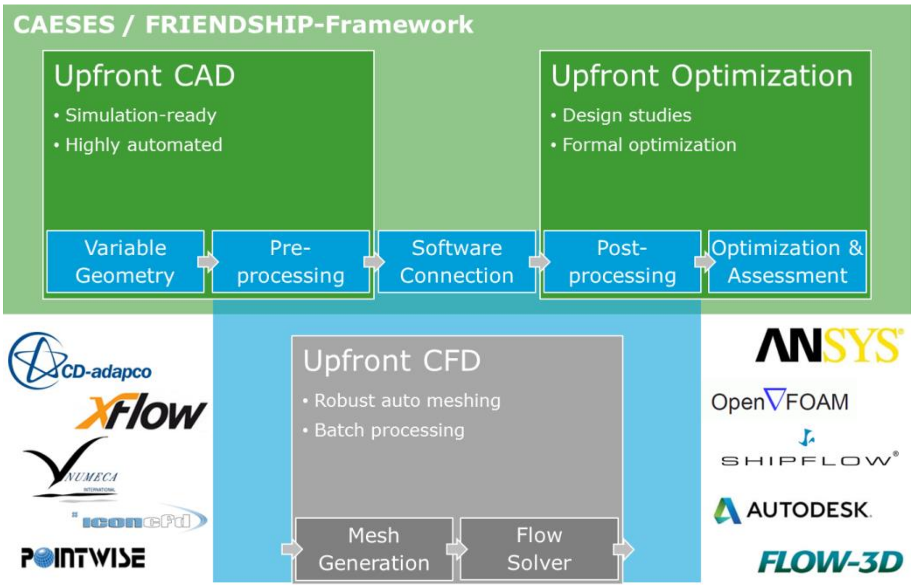

Fig. A-2: CAESES® / FRIENDSHIP-Framework for practical optimization using external CFD codes