

Advances in Computational Geometry and Design Optimization

Rohan K. Patel, Akira S. Nakamura,
Ethan R. Lee, and Lucas F. Jensen

Rohan K. Patel and Akira S. Nakamura are affiliated with the Department of Mechanical Engineering, University of California, Berkeley, CA, USA.

Abstract

This paper presents an approach which increases the flexibility of a computer-aided design (CAD) model by automatically refining its parameterization and adding new CAD features to the model's feature tree. It aims to overcome the limitations imposed by the choice of parameters used during the initial model creation, which constrains how the model shape can change during design optimisation. Parametric effectiveness compares the maximum performance improvement that can be achieved using a parameterisation strategy to the maximum performance improvement that can be obtained where the model is unconstrained in how it moves. As such, it provides a measure of how good a parameterisation strategy is and allows different strategies to be compared. The change in parametric effectiveness due to inserting multiple different CAD features can be calculated using a single adjoint analysis; therefore, the computational cost is essentially independent of the number of parameterisation strategies being analysed. The described approach can be used to automatically add new features to the model and subsequently allows the use of the newly added parameters, along with the existing parameters to be used for optimization, providing the opportunity for a

Indexed keywords: cad . feature . optimisation . adjoint . parameterisation

Article History: Received: 17 April 2025 | Accepted: 17 June 2025
| Published: 07 July 2025

better performing product. The developed approach is applied on CAD models created in CATIA V5 for 2D and

3D finite element and computational fluid dynamic problems.

1 Introduction

Parametrisation is at the core of optimisation as it defines the design space the optimising algorithm explores. The success of any shape optimisation methodology depends extensively on the type of parameterisation technique employed (Samareh 2001). One straightforward route, which gives the most flexible parameterisation strategy, is to use the nodes of the computational mesh (Belegundu and Rajan 1988; Jameson et al. 1998; Helgason and Krajnovic 2012; Jameson 2004) as design

Responsible editor: Nam Ho Kim

* Trevor T. Robinson
t.robinson@qub.ac.uk

- ¹ School of Mechanical and Aerospace Engineering, Queen's University Belfast, Belfast BT9 5AH, Northern Ireland, UK
- ² School of Mechanical and Maritime Engineering, Liverpool John Moores University, Liverpool, UK
- ³ Volkswagen AG, Group Research, CAE Methods, Wolfsburg, Germany

Published online: 28 November 2018

variables. One major drawback for this parameterisation strategy is that, as all surface mesh nodes can move independently, the implementation of a post-optimisation smoothing algorithm is required to prevent the appearance of non-smooth shapes during the optimisation process. In this regard freeform deformation techniques have been successfully implemented for aerodynamic shape optimisation problems (Samareh 2004; Kenway et al. 2010; Hoogervorst and Elham 2017). These techniques originated from the soft object animation in the computer graphics industry (Sederberg and Parry 1986). The benefit of this approach is that it imparts smooth deformations to the analysis mesh and enables the parameterisation to alter the thickness, sweep, twist, etc. for the design of an aerospace system. One of the drawbacks of these mesh-based



© 2025 The Authors. Open Access under CC BY 4.0.

How to cite: Rohan K. Patel, Akira S. Nakamura, Ethan R. Lee, and Lucas F. Jensen (2025). Advances in Computational Geometry and Design Optimization. Journal of Computer Engineering, 14(7), 117–138. DOI: <https://doi.org/10.5281/zenodo.19357905>

optimisation methods is that the mesh topology (in terms of the number of elements present and their connectivity) must remain constant as the model updates. Also, it is the mesh that reaches the optimum shape. As most manufacturing processes require a CAD model, this mesh must then be translated into a CAD model before it can be used for further analysis or manufacturing assessments. This mesh-to-CAD step is non-trivial and may require extensive user interaction (Guo 1997; Anderson and Crawford-Hines 2000). It is also very difficult to match the shape directly.

Using CAD geometry within the optimisation process directly removes a lot of these difficult steps and should result in a better-quality model at the end of the process. It also aligns with the industrial ambition of having a more integrated design process. In a feature-based CAD modelling system (e.g., CATIA V5 (2017)), a part model is comprised of individual features which are combined to represent an overall shape (see Fig. 1). In the context of CAD modelling, features can be defined in many ways depending on their application (Shah 1991; Latif et al. 1993; Salomons et al. 1993). In this work, a feature is any operator in the CAD model feature tree that can be manipulated to change the model's shape, and a parameter is the dimensions associated with the feature. For example, these might control the shape of the sketches on which sketch-based features (e.g., loft) are constructed, or the length of the feature (e.g., an extruded boss).

When creating a CAD model within a CAD modelling system, it is common to create relationships to specify that the value of one parameter is a function of the values of other parameters in the model. This relationship is part of the design intent of the model, and once applied the parameters they relate to cannot be controlled independently. The parameters which are responsible for changing the model's shape can have real, integer and Boolean values. These parameters can be accessed using a suitable CAD system application programming interface (API). In this paper, only continuous dimensional parameters which affect the shape of the model are considered. Modifying parameters with real values (such as lengths and angles) by small amounts will typically cause movements of the boundary proportional to, and of the same order as, the size of the parameter perturbation. In CATIA V5, the CAD features are stored in a feature tree as shown in Fig. 1a and a list of all of the parameters used to construct the model can be found through the "Formulas" window as seen in Fig. 1b.

Some authors (Martin et al. 2012; Yu et al. 2011; Xu et al. 2014, 2017; Burgos 2015) have attempted to develop a link between optimisation processes and CAD models through the use of non-uniform rational B-splines (NURBS) patches. A NURBS patch can be modified by moving its control points and these can be used as optimisation variables. Here, the benefit is that the NURBS can represent a wide variety of shapes and represents a richer design space than that can be expressed using parameters in a feature-based CAD model. However, one downside of using NURBS is that sometimes the NURBS control net may be too coarse in certain regions and requires a process to enrich the control net by adding more control points before it is used for optimisation. Jesudasan et al. (2017) presented an adaptive parameterisation approach based on NURBS patches, where the NURBS control net was refined by using knot insertion and subsequently used to optimise pressure loss across a U-bend passage of a turbine blade serpentine cooling passage. The downside of the approaches outlined above is that, as they do not work directly on the parametric CAD model created in a feature-based CAD system, the design intent and the parametric associativity captured in the choice of features used to build the model is lost.

If there is a desire to use CAD model parameters as variables for optimisation, one of the key issues is that there is no clear link between the CAD parameters and the effect they have on performance, although (Agarwal et al. 2018) have shown an approach based on adjoint sensitivities which enables this. However, the ability to optimise the model is limited by the choice of CAD features in the model, and sometimes the parameters used to define these features may not be the best choice for optimisation (in many instances they are not selected with optimisation in mind). In this case, it is necessary either to re-parameterise the existing features (i.e., use a different way to control shape) or add more features (each with its own set of parameters) to increase the flexibility of the model. This is not straightforward, especially if the person implementing these modifications is not the initial creator of the CAD model and is not fully aware of how it was constructed.

The adjoint surface sensitivities represent the derivative of the objective function with respect to the normal displacement of the surface mesh nodes. As such, the distribution of adjoint sensitivity over the surface of the model indicates how the boundary of the model should move to achieve an optimum improvement in performance. The advantage of the adjoint method arises from the fact that the gradient computation is independent of the number of design variables and so in the context of this work, many parameterisation strategies can be evaluated cheaply. The adjoint is computed by solving a problem of computational cost comparable to that of the primal analysis. A detailed description of the adjoint-based approach can be found in (Giles and Pierce 2000;

automatically. It achieves this by adding new features to the CAD model when the parametric effectiveness is shown to be low. Thereafter, the full set of CAD parameters (new and original) are used to change the shape of the model. In this work, this is achieved by updating the CAD model in two ways: (1) the sketch entities are modified, specifically straight lines are converted to splines and splines have additional control points added, and (2) by adding new sketch profiles to the definition of multi-section solid features that are common in both aerospace and automotive designs. The insertion of new features is done directly into the CAD model feature tree, so the model can be easily used for downstream applications like manufacturing or process planning without any

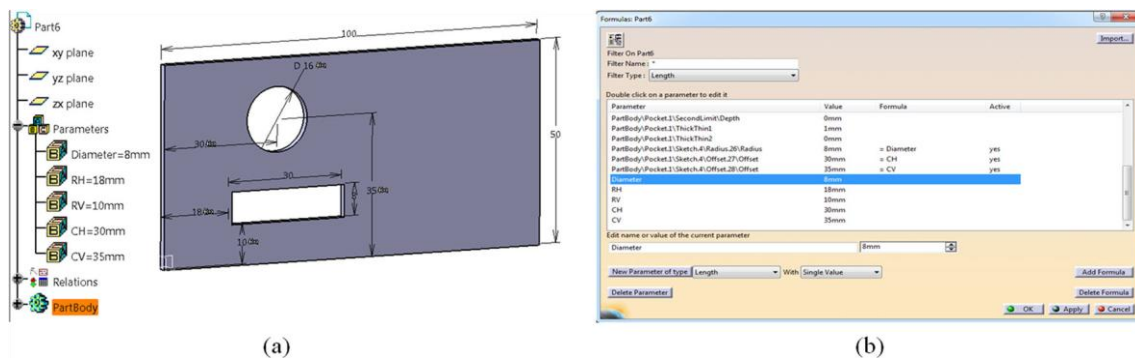


Fig. 1 CATIA V5 feature tree representation

Nadarajah and Jameson 2001; Othmer 2008; Mader et al. 2008). The adjoint surface sensitivities can be integrated within a CAD-based optimisation framework by calculating the parametric design velocity, i.e., the boundary shape movement resulting from a change in CAD parameter.

Parametric effectiveness (Robinson et al. 2013) rates the ability of the parameters in a model to change its shape in an optimum way. Here, the optimum shape change is assumed to be that suggested by an adjoint sensitivity map on the model boundary. It ranges from 0 to 1. A low parametric effectiveness indicates that the parameters in the model are not able to perturb the CAD model shape in the manner the adjoint sensitivity map suggests. In this work, this is taken as an indication that it is better to re-parametrise or add new features to the model before optimisation. Robinson et al. (2012) defined the basic strategies for calculating the change in performance which could be achieved by the addition of small structural features including stiffener, boss, hole and pocket. But, the authors did not detail how this could be used to automate the process as part of a general design strategy.

In this work, a framework is presented which increases the design flexibility of a CAD model

additional post-processing relative to a standard model. It will be argued here that the design decision on which CAD feature to insert, and where, can be made by analysing the adjoint sensitivities. Further, a comparison of parametric effectiveness and the optimisation results for a number of CAD models is presented where different updating strategies are applied.

In this work, the CAD models are created in CATIA V5 and optimised using sequential least square programming (SLSQP) method implemented in Scipy (Scipy Optimize 2017). SLSQP is a gradient-based optimisation algorithm which minimises a function with any combination of bounds, equality and inequality constraints. The advantage of using this algorithm is that it avoids the need for determining the weighting factors by including constraints directly into the optimisation problem. Automated CAD system APIs are developed for refining the CAD features in Python 3.5 (2017), and an interface with CATIA V5 is created using win32com object programming (Hammond and Robinson 2000).

The remainder of the paper will first summarise the integration of the adjoint-based sensitivity calculation with the design velocity. Section 3 presents the

methodologies proposed for refining the CAD features and also inserting new features into the CAD model feature tree; the developments are demonstrated for the optimisation of a loaded cantilever beam in Section 4 and an automotive duct in Section 5; the proposed methodology and the results are discussed in Section 6; and the paper finishes with the main conclusions.

2 Theory

2.1 Design velocity

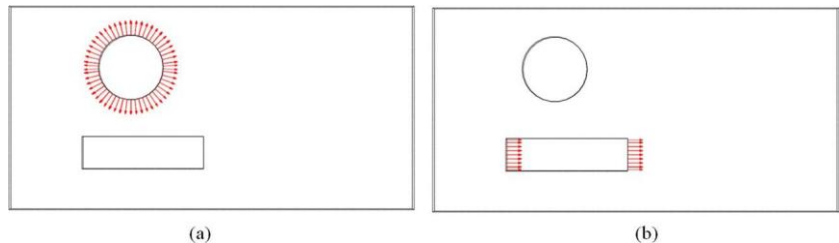
A key requirement when using CAD models for optimisation is the understanding of how the parameters defining the CAD model affect its shape. This is not always obvious from the CAD model feature tree, where the choice of modelling strategy is highly dependent on the preferences of the person who created the model. Design velocity (V_n) is the normal component of the displacement of the boundary of a CAD model. This work is concerned with computing the design velocity caused by the perturbations of the CAD parameters and is computed as

$$V_n = \frac{1}{\delta t} \delta X_s \cdot \hat{n}$$

where δX_s is the movement of surface at a point on the boundary, and \hat{n} is the outward unit normal of the surface at that point. Now, considering the CAD model in Fig. 1, the parameters controlling the diameter of the hole and horizontal position of the slot are perturbed, and the design velocities are computed to represent the movement of the boundary in response, as shown in Fig. 2. The convention adopted throughout this work is that a positive design velocity represents an outward movement of the boundary, and negative is inward. Therefore, the hole diameter and right edge of the slot exhibit a negative design velocity in Fig. 2, whereas the design velocity at the left edge of the slot is positive.

A number of approaches to calculate design velocity

Fig. 2 Design velocities for parameters shown in Fig. 1. a Changing hole diameter. b Horizontal movement of slot



have been proposed in the literature (Kripac 1997; Chen and Tortorelli 1997; Hardee et al. 1999), but are usually restricted to geometric models where the

boundary topology remains constant through the parametric perturbation. One of the ways to compute design velocities is by directly differentiating the CAD system using automatic differentiation (AD). Xu et al. (2014) differentiated an in-house CAD tool based on NURBS patches by using the AD tool tapenade developed by INRIA Sophia-Antipolis (TAPENADE 2018). Torreguitart et al. (2017) presented the differentiation of another in-house CAD tool by using AD software tool developed at the University of Paderborn (Walther and Griewank 2010). Recently, (Auriemma et al. 2016; Banovic et al. 2017) demonstrated the differentiation of the open-source CAD system Open-CASCADE (v7.0). These approaches to calculate design velocity have significant advantages as they do not require a geometry or mesh to be recomputed. They are both efficient and robust against BREP boundary topology changes. These approaches also avoid the difficulties with numerical accuracy associated with finite difference approaches. The downside is that it requires access to the underlying source code of the CAD system, which is unlikely to become an industrial reality for the major CAD systems in the near future. Further, the derivative calculation for complex geometric configurations and Boolean operations is still challenging.

The calculation of design velocity can be complex for industrial models, where due to the nature of the boundary representation produced by the CAD system, the shape changes caused by a parameter perturbation can be highly constrained, or the model can suffer from robustness issues. The approach used in this work for calculating design velocity is detailed in Agarwal et al. (2018), where the design velocities were calculated for CAD models of industrial complexity and where the parameters defining the CAD model were used as design variables. The approach is based on the faceted approximations of CAD model geometry generated using open-source mesh generator GMSH (Geuzaine and Remacle 2009). This method is applicable to any feature-based CAD modelling package, places no constraints on changes to boundary topology caused

by parametric changes and does not require any special access to the source code of the CAD kernel.

2.2 Adjoint method

In this paper, adjoint sensitivity maps are used to gain insight into how the movement of the boundary of the model can impact the objective function. The adjoint sensitivity maps suggest the direction that the boundary of the model should be moved and the relative magnitudes of the movement which should be applied over the boundary, to improve performance. This work contributes no further enhancements to the continuous adjoint method and, thus, a detailed description of the adjoint formulation is omitted. More information about the adjoint development can be found in the literature, e.g., (Papoutsis-Kiachagias and Giannakoglou 2016). The adjoint surface sensitivity (ϕ) represents the gradient of objective function with respect to the normal displacement of the surface mesh nodes and can be mathematically defined as $dJ = \int_A \phi \delta X_s dA$

where J represents the objective function, and δX_s is the movement of surface in the normal direction at a point on the boundary. The computation of adjoint

Fig. 3 2D line converted to spline with two control points coincident with end points of the line

sensitivity requires an additional analysis, where the field adjoint equations are solved. This is done at a computational cost practically independent of the design variables, which is the main advantage of this method, especially where there are a large number of parameters to be considered, such as in this case when working with a feature-based CAD model. Using the computed primal and adjoint fields, the adjoint sensitivity map is computed, which shows how the geometry shape should change, in order to achieve a reduction in the objective function. The adjoint surface sensitivity ϕ is provided by the computational analysis package. In this work, the primal analysis is performed with OpenFOAM®, whereas the system of adjoint equations is solved using the Helyx adjoint solver developed by Engys (Karpouzas et al. 2016).

2.3 Parametric sensitivity and effectiveness

Parametric sensitivity (S) is the change in objective function caused by an infinitesimal perturbation of a CAD parameter. For a computer-aided engineering model, the total change in objective function (dJ) due to parametric perturbation can be predicted as the summation over the boundary of

$$dJ = \int_A \phi \delta X_s dA \quad \delta J = \int_A \phi \delta X_s dA$$

where ϕ is the surface sensitivity for the objective function, and A represents the surface area of the boundary. Knowing the change in objective function due to the parametric perturbation in question, the parametric sensitivity can be calculated by normalising this value with respect to the size of the perturbation as

$$S = \frac{dJ}{d\theta} = \int_A \phi \delta X_s dA / d\theta$$

where $d\theta$ is the perturbation of a CAD parameter.

Parametric effectiveness (Robinson et al. 2013) compares the maximum change in performance that can be achieved using the CAD model parameterisation to the maximum performance improvement that can be achieved if the model is not constrained by the parameterisation, where both are subjected to the constraint of a unit root-mean-squared boundary movement. The imposed constraint of boundary movement ensures that a parameter moving an area of low sensitivity by a large amount

would not be favoured compared to a parameter causing a small localised movement in the areas of high sensitivity. The detailed mathematical derivation of the measure can be found in Robinson et al. (2013). The parametric effectiveness can be computed for different combinations of CAD parameters using a single adjoint analysis. This allows the designers to select the appropriate set of parameters to be used for optimisation.

Herein, an automated approach is developed to compute the parametric effectiveness of CAD model parameters. In cases where the parametric effectiveness is low, a novel methodology is shown which automatically adds the optimum features to the CAD model feature tree and thus increases the design freedom of the model. The change in the value of parametric effectiveness is used as a criterion to decide which set of the newly inserted features provides the greatest performance improvement or to compare different parameterisation options.

3 Methodology for feature insertion

For a CAD model which is determined to have a low parametric effectiveness, this paper describes three approaches to increase its parametric effectiveness through adding new features and associated parameters to the CAD model feature tree. Sections 3.1 and 3.2 focus on refining 2D sketches (adding more parameters) on which 3D CAD features are built (e.g., extruded bosses, revolves, etc.). Section 3.3 is an enhancement in the definition of the solid feature by adding additional profile sketches to an existing CAD feature. 3.1 Replacing a 2D straight line with a spline

A straight line is a geometric entity with zero curvature, defined only by its start and end points. By contrast, the shape of a spline curve is defined by the position of its control points, which at a minimum must consist of a start and end point, but for which there is no upper limit on the number there can be. Where there are only start and end control points (and no additional tangency information), a spline curve will be equivalent to a straight line between those points. For a sketch, the automated approach described herein will replace all lines in the sketch with spline curves, where the start and end points of the line are used to define the spline, as shown in Fig. 3. If the positions of the start and end points of the spline are not parameterised, then the approach will also add parameters to control their position (a horizontal and vertical parameter). This is implemented for a parametric CAD model constructed in CATIAV5 using the CAD system API. It should be noted that replacing a line with a spline controlled by two control points does not increase the flexibility of the model, which comes in the next step.

3.2 Adding additional control points to 2D spline

Increasing the number of control points defining a spline curve increases the design freedom in the curve and allows it to generate more varied shapes. The next step in the process is to automatically insert additional control points into every spline curve in the sketches. Here, the control points are added at the midpoint of

Fig. 4 Additional control points inserted in 2D spline

each section of the spline. This means that in the first iteration, the control point will be added at $t = 0.5$ along the spline, creating two spline sections. In the next iteration, control points will be added to the

midpoint of each spline section ($t = 0.25$ and $t = 0.75$ along the original spline) and so on. Parameters are added to control the values of the x and y coordinates of the additional control points inserted, as shown in Fig. 4. The insertion of the control points onto the spline curve alone does not change the model shape. However, when the positions of any of the control points are changed during the optimisation process, the shape of the spline will update, and any solid modelling feature based upon that sketch will also change in shape accordingly. As such, these new control point parameters give added extra flexibility to the model, and perturbing each will generate a design velocity on the model boundary. Now, these design velocities are linked with the adjoint sensitivities to obtain the parametric effectiveness for the addition of each control point. The addition of control points to the model can be repeated many times, and in this work, the choice of whether or not to continue adding more control points is based on the improvement in parametric effectiveness they achieve.

3.3 Modifying a multi-section solid

Many real-world components vary in profile along their length (e.g., an aircraft wing or an automotive duct). These components are usually modelled in a CAD system by creating sketches at different positions and orientations along the length of the components and then creating a multi-section solid CAD feature (sometimes referred to as a loft) passing through these profiles. Figure 5 shows a simple multi-section solid feature built-in CATIAV5. These features typically pass through the sketches/profiles exactly but then interpolate the shape of the feature between the sketches. Guide curves are optional curves that run along the length of the multi-section feature, joining the different profiles and guiding the interpolation that occurs between them.

Changing the shape of the underlying sketches will change the shape of the resulting feature. One way to update the feature is to use the strategies described in Sections 3.1 and 3.2 to update the shapes of the sketches from which it takes its shape. The other way is by inserting additional 2D sketches along the length of the feature and using the parameters associated with

the shape of the new sketch as design variables. The greatest benefit will be obtained from the optimisation process if the parameters are added in the areas of high adjoint sensitivities. Once the position to add the

sketch is selected (which could be done automatically based on the surface sensitivity, but in this work is done manually), the addition of the new sketches was implemented using the following automated process:

Fig. 5 Multi-section solid feature in CATIA V5

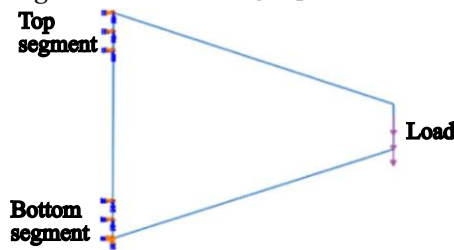
Fig. 6 a A multi-section solid with a point on the surface. b Updated multi-section solid with new sketch created passing through the selected point

1. surface (point with highest adjoint sensitivity) as shown in Fig. 6a.
2. Two planes are created passing through profiles at the start and end of this feature. These two planes are intersected, creating an intersection line. A new plane is created which is coincident to both the computed intersection line and the point of highest sensitivity. In the special case where the planes created at the start and end of the multi-section solid are parallel, then the new plane is created which is parallel to the start plane and is coincident to the selected point.
3. The new plane is intersected with the multi-section feature to get a boundary profile.
4. A sketch is then created on the plane to match the existing profile shape using vertex and edges of the boundary profile as a reference.
5. If the selected multi-section feature includes the guide curves, the definition of guides is modified to include the sketch vertex in closest proximity.
6. Finally, the new sketch profile created in step 4 is included in the definition of multi-section feature by recreating the feature to include the old and new profiles as shown in Fig. 6b.

By following the steps in Sections 3.1, 3.2 and 3.3, any sketch-based features in the model should now be based on sketches controlled by more parameters than before, and any multi-section features should be defined using more sketches than before. The new features and parameters will have been added because they increase the parametric effectiveness of the model. It is important to note that at this stage, the shape of the CAD model will not have changed. While

Fig. 7 a Design domain of cantilever beam with boundary conditions. b Optimisation constraints

Fig. 8 a CAD 2D sketch with lines. b CAD 2D sketch with lines transformed to splines



1. Select the multi-section feature (containing region of high adjoint sensitivity) to be updated and define a point on its

the additional features and parameters will add additional control flexibility to the model, their current values should be set such that the model shape is the same as the original part. Changing these new control points during the optimisation process will produce design velocities unobtainable with the initial parameter set.

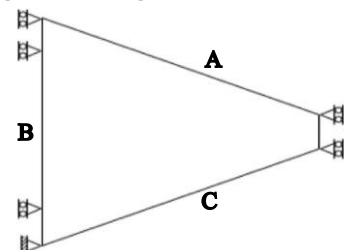
4 Example application: cantilever beam

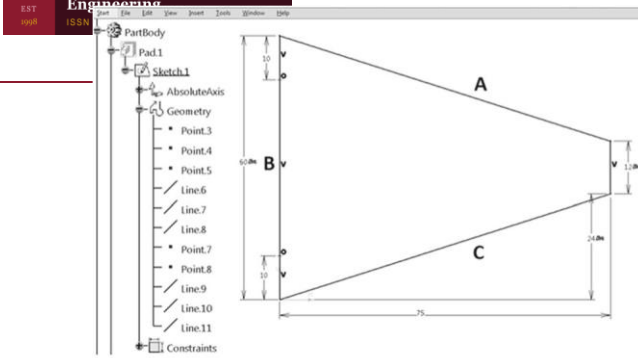
A loaded cantilever beam model is the first test case used to demonstrate a methodology for refining the CAD parameterisation during the optimisation. The design domain is shown in Fig. 7a, and the beam is loaded with a traction force at the right-hand edge. Top and bottom segments have been isolated on the left side and are fixed so that they cannot move in the horizontal direction.

Figure 7b shows the constraints on the mechanical model. The length of the top and bottom segment of the left edge and the length of the right edge are fixed. The bottom left corner of the beam cannot move. Thus, only the sketch lines labelled A, B and C in Fig. 7b can be modified during the optimisation. The optimisation problem considered here is the minimisation of strain energy of the system with the constraint that the volume is fixed throughout the optimisation. This can be described as

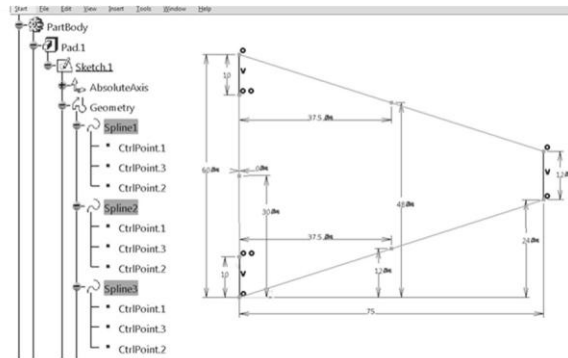
$$\begin{aligned} \text{Minimise :} & \quad \text{strain energy} \\ \text{Subject to :} & \quad \text{change in volume } \frac{1}{4} \quad \delta P = 5 \end{aligned}$$

Design variables : vector of CAD parameters
 Design space : 2:5 length of left edge of the beam





(a)



(b)

The optimisation is a compliance minimisation problem; therefore, the objective function for this test case is to minimise the strain energy. This type of problem is selfadjoint, meaning that a special adjoint solver is not required to compute the surface sensitivities, as the contours of strain energy density on the surface of the model indicate the change in strain energy in the component that can be achieved by moving the boundary. The CAD model is constructed in CATIA V5 as an extruded boss from a sketch of the cross section (Fig. 8). Updating the sketch will therefore update the entire part. The finite-element mesh generation and analysis of the model are done in ABAQUS CAE, which is linked with the CAD system and Scipy optimisation modules using python scripts. The constraint of the constant volume was imposed by

defining a parameter (vol) in CATIA V5 which returns the volume of the solid as a measurement value. A series of CAD system APIs are developed in this work to automatically obtain the value of vol parameter from the CAD system. The computation of gradient of the constraint function is performed by using an API to perturb the CAD parameters by a small amount ($\Delta\theta$) and obtain the new value of the vol parameter, and use finite difference method to obtain the gradients as

$$\text{gradient} \frac{\partial \text{vol}}{\partial \theta} = \frac{\text{vol}_{\text{perturb}} - \text{vol}_{\text{initial}}}{\Delta\theta}$$

The model was created as an extruded boss feature

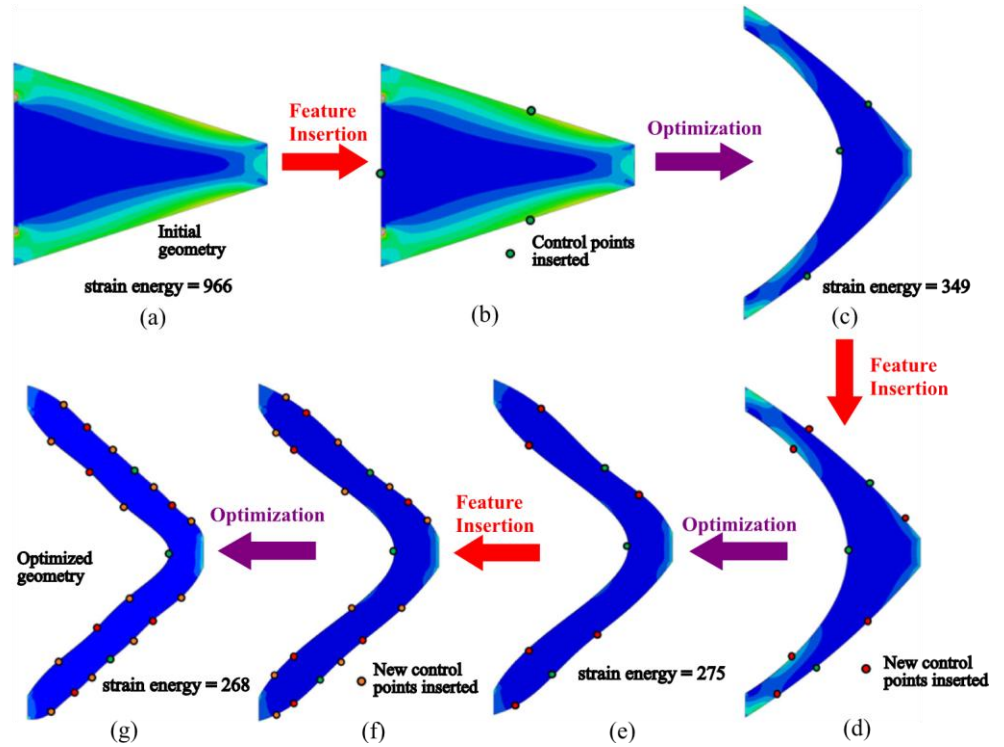
to 0.16 and a total strain energy of 349 (~64% reduction). Once the optimiser terminates, additional spline control points are automatically added to the splines using a Python CAD

(pad in CATIA V5). This consisted of designing the

cross-section Fig. 10 CAD model of S-bend duct created in

CATIA V5

Fig. 9 Adaptively inserting CAD features during optimisation



which was extruded to create the beam. The design parameters of the initial model are shown in Fig. 8a, for which the strain energy contours are shown in Fig. 9a, and the total strain energy is calculated as 966. The parametric effectiveness computed for the initial model configuration is 0.0 which indicates that the component cannot be optimised without violating the constraints. Now, the parameterisation of the model was updated using the methodology outlined in Section 3. Here, the approach was used to adaptively refine the model meaning that when new features or parameters were added, an optimisation step was carried out and terminated before more new features or parameters were added. In all cases, the new features and parameters were visible in the CAD model feature tree. The constant volume constraint is used to ensure the model did not grow indefinitely as an optimisation with the objective of reducing strain energy will attempt to do. The gradient of this

constraint is computed as the change in volume of the beam (computed within CATIA V5) with every parametric update.

The refinement of the model was as follows: Firstly, the straight lines are transformed to splines each with a control point at its centre using the process described in Sections 3.1 and 3.2 as shown in Fig. 8b, resulting in a parametric effectiveness to 0.24. The position of new control points is also shown in Fig. 9b. The model is then optimised using SLSQP algorithm with the constraint of constant volume, resulting in a CAD geometry shown in Fig. 9c with effectiveness reduced

system API (shown in Fig. 9d) to increase the parametric effectiveness to 0.43.

A second stage of optimisation was then run with the new set of parameters resulting in the model shown in Fig. 9e with the total strain energy of 275 (~71.5% reduction overall) and with the parametric effectiveness reduced to 0.15. Now, additional spline

control points were added to increase the effectiveness to 0.35 as shown in Fig. 9f and a third stage of optimisation is performed. At this stage, the optimised model (Fig. 9g) possessed strain energy of 268 (~72.3% reduction) with an effectiveness of 0.17. Since a very small gain in performance was achieved during the last optimisation step, the insertion of new

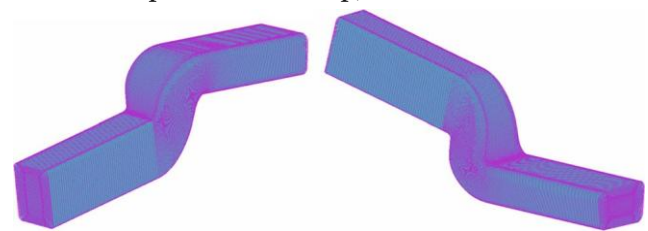
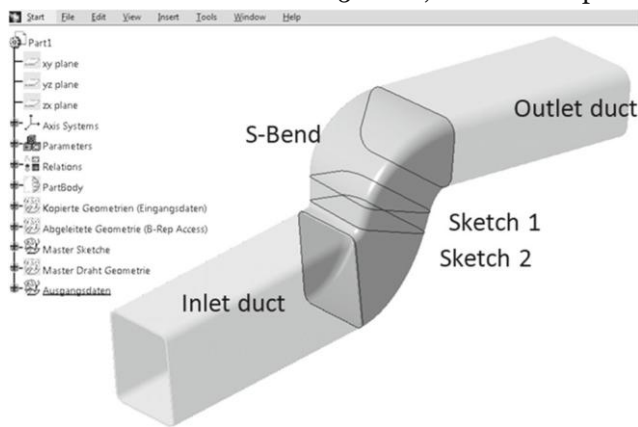


Fig. 12 CFD analysis mesh for S-bend

features was stopped at this stage. It is evident that the optimised design is similar to the optimisation results obtained in (Mei et al. 2008; Wang et al. 2003). The process of adding more control points to the spline curve takes less than 5 s, while the process of generating the new mesh and running an FEM analysis within ABAQUS takes 25–35 s on a 3.60GHz workstation with 16GB RAM. The overall optimisation process including the modification of the CAD model and the reduction in strain energy is visualised in Fig. 9.

5 Example application: S-bend

The parametric CAD model of the S-bend duct shown in Fig. 10 was created in CATIA V5. It was modelled using 2D sketch profiles at different positions and orientations, and then by creating a multi-section solid passing through these

Fig. 11 S-bend CAD parameterisation

Fig. 13 Pressure contours (top) and velocity contours and streamlines (bottom) computed for the base geometry profiles. The duct is composed of three individual sections, i.e., inlet, S-bend and outlet. The construction of the multisection solid feature uses guide curves to control the orientation and curvature of the outer boundary, particularly for the S-bend area. As the inlet and outlet ducts join with other components, their shape is fixed and they are not considered for optimisation. This means that the only geometrical features that can be varied are the two sketches (sketch 1 and sketch 2) describing the profile of the S-bend at different positions along its length.

The two sketches are parameterised using the parameters as shown in Fig. 11, while the orientation of the sketch planes is defined by an angle parameter relative to the inlet plane. The fluid domain used for CFD analysis is extracted from the body within the CAD environment by joining the face patches with tangency constraints. This imposes C_1 continuity between the wall faces.

The flow is laminar, with Reynolds number $Re = 350$, calculated with a hydraulic diameter of $D_h = 0.053\text{m}$, inlet velocity of $u = 0.1\text{m/s}$ and kinematic viscosity of $\nu = 1.511 \times 10^{-5}\text{m}^2/\text{s}$. The CFD mesh created in ICEM CFD (Ansys 2017) comprises 486,000 elements and is shown in Fig. 12. The results of flow analysis are shown in Fig. 13,

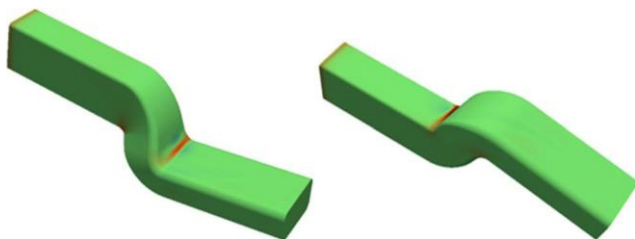
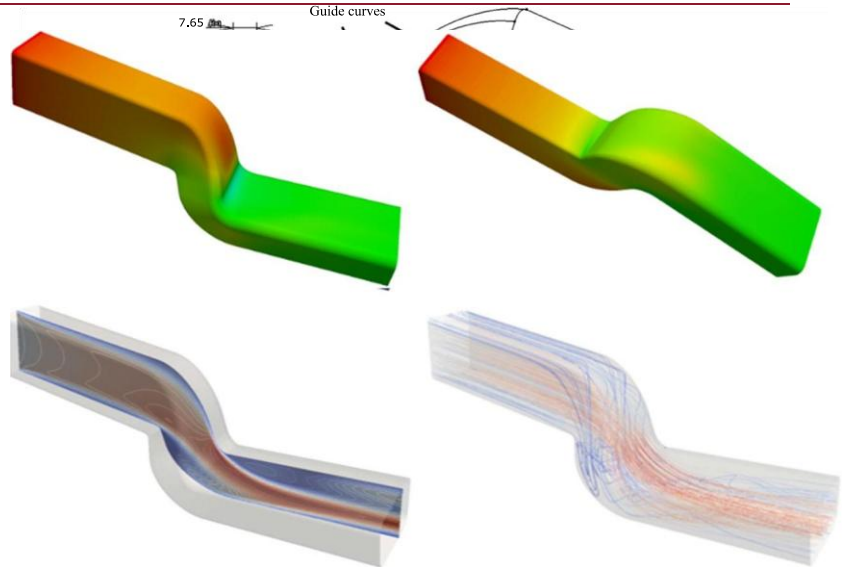


Fig. 14 Adjoint sensitivities contour. To minimise the objective function (dissipated power), the surface should be pulled out at positive values (warm colours) or pushed in (cold colours) where the pressure and velocity contours are plotted along with streamlines.



The objective function considered for optimisation is the power dissipation through the duct (Othmer 2008), defined as

$$J = \frac{1}{2} \int_{\Delta V} \mathbf{n} \cdot \mathbf{p} \, dA - \frac{1}{2} \int_{\Delta V} \mathbf{v}^2 \, dV \quad (7)$$

where p and v are the pressure and velocity of the flow, and A and \mathbf{n} are the surface area of the duct and the unit normal of the duct pointing away from the fluid area.

In this example, the adjoint was computed as a separate analysis step using Helyx (Karpouzias et al. 2016). The adjoint sensitivities of the objective function with respect to the normal displacement of the surface are computed at the first optimisation cycle and presented in Fig. 14. High sensitivities can be seen at the areas of the duct with high curvature. These are the areas in Fig. 14 where flow separation occurs, contributing to an increase in power dissipation. The adjoint sensitivities indicate the geometry displacement that would control the flow recirculation and reduce the objective function. It is clear though, the changes suggested by the adjoint sensitivities are local and cannot be implemented easily by the original design parameters (Fig. 11). The value of parametric effectiveness calculated for the initial configuration is 0.14, which is taken here as an indication of the need to add new features to the model before optimisation.

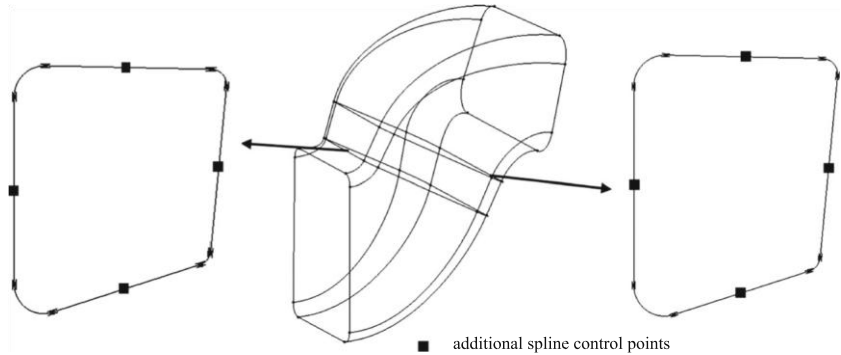


Again, this is achieved automatically following the methodology described in Section 3.

5.1 Replacing 2D lines with splines with added flexibility

The first step to increase the design flexibility of the S-bend model is to replace straight lines in existing

Fig. 15 Replacing lines with splines with additional control points



sketches and parameterising the positions of the end points. Figure 15 shows that the straight lines in the two sketch profiles (initially parameterised as shown in Fig. 11) are converted to spline curves, and spline control points have been added at the centre of each spline curve. In this case, the radius of circular arcs (defining the rounded corners) and their tangent with the lines are formulated as constraints. A total of eight design parameters are used (in addition to the radii of corner circles and tangency between lines and circles) to make the sketch fully constrained. This results in a total of 32 parameters with a parametric effectiveness of 0.22.

5.2 Addition of multiple cross-section sketches

Since the duct is defined as a multi-section solid, the process described in Section 3.3 was used to insert new 2D sketch profiles along the length of the duct, and the parameters associated with new sketches used as design variables. It is apparent that the best location to add sketches would be in the areas of high adjoint sensitivity, increasing the design flexibility in these

areas. Analysing the adjoint sensitivity map in Fig. 14, the two multi-section features defining the two bend areas of the duct are user selected as a point on the surface of each feature in the region of high adjoint sensitivity. The new sketch profiles A and D are inserted using the automated process described in section 3.3 and are shown in (Fig. 16). The addition of sketch A resulted in an increase in parametric effectiveness to 0.57, and then adding sketch D

increased the effectiveness to 0.61 (total number of parameters increased to 48). Further, using the methodology of Section 5.1 to convert the straight lines of new sketches to spline curves with additional control points resulted in a very small increase in the parametric effectiveness, and thus, it was decided that with a parametric effectiveness of 0.61, the optimisation should be attempted.

5.3 Optimisation with new parameters/features

The effect of the increased value of parametric effectiveness is demonstrated using the design velocity contours comparing the movement of the CAD model using the steepest decent boundary movement based on the initial parameterisation (selected by designer) with the steepest decent boundary movement of the parameterisation created by inserting features into the CAD model to get the maximum parametric effectiveness. It can be seen in the design velocity contours in Fig. 17b how the new features allow the movement of CAD model

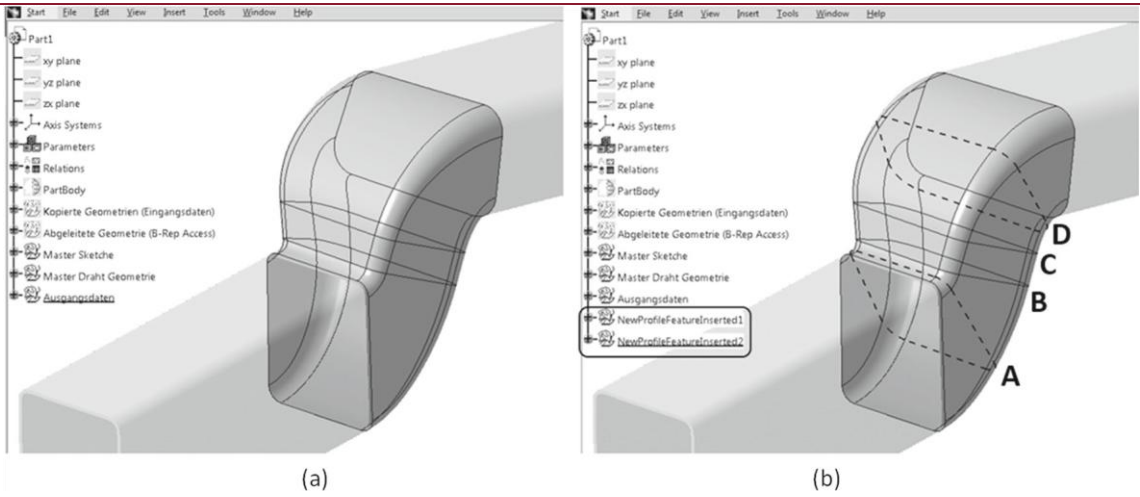
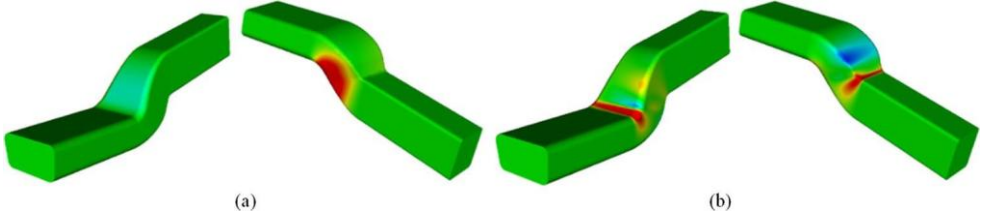


Fig. 16 Insertion of new cross-section sketches in the CAD feature tree. a Original CAD. b CAD model with new sketches shown with broken lines Fig. 17 Design velocity contours (red as positive and blue as negative) for the overall boundary movement $dV = 1 E^{-4}$. a Initial parameters. b With feature insertion



boundary in the areas close to the optimum movement suggested by the contours of the adjoint sensitivity map in Fig. 14, compared to Fig. 17a which shows much less movement localised in the highly sensitive region given by the original parameters.

In this work, the optimisation was performed using different parameterisations. The first was the one selected by the designers, and the others were created by inserting new features into the CAD model feature tree using the described approach. The insertion of new features was done in three different ways: (1) adaptive parameterisation where each time a new feature is added to the model and an optimisation was carried out before additional features are added, (2) progressive parameterisation where it was determined upfront that new features should be added to the model after fixed number of optimisation steps, and (3) static parameterisation, where as many new features were added to the model before the start of optimisation as were required to improve the parametric effectiveness as much as possible. The results of optimisation with an objective of minimising power-loss performed using different methodologies are shown in Fig. 18. A gradient-based optimisation method (steepest decent) is employed to minimise the

objective function using the CAD parameters as design variables. With each optimisation step, the CAD parameter values are updated using a CAD system API developed during this work, and for each step, a new analysis mesh was created.

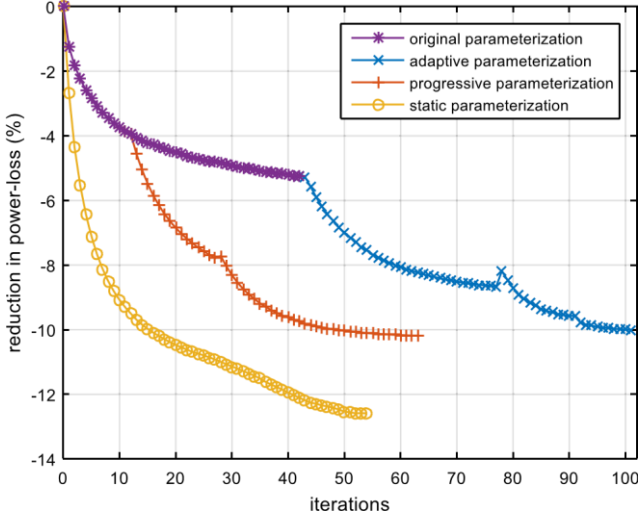


Fig. 18 Optimisation with CAD features inserted in S-bend (power-loss as objective function)

The objective function was reduced by 5.25% in 42 steps when optimised using the initial set of design parameters. Following the adaptive and progressive

parameterisation strategy, the objective function was reduced by 10% in 102 and 63 optimisation steps respectively. Using the static parameterisation strategy, the maximum reduction in objective function was 12.6% and required only 54 steps. The computational cost for the optimisation processes is outlined in Table 1.

Jesudasan et al. (2017) presented an adaptive parameterisation strategy and showed that higher reduction in objective function can be achieved by adaptively refining the NURBS patch using knot insertion during the optimisation. In this work, it was found that refining the CAD model parameterisation at the start of optimisation (static parameterisation) was superior to the other re-parameterisation approaches in terms of optimisation time and improvement in performance achieved. In the remaining paper, the optimisation results are shown for the static parameterisation. The evolution of parametric effectiveness during the optimisation process is shown in Fig. 19. The parametric effectiveness tends to a small value as the geometry reaches its optimum shape. This is because the parameters have become less able to improve performance as they tend to their optimum value.

The comparison between the initial and optimised CAD model is shown in Fig. 20. The optimised design shows some ripples in the shape near the regions where the S-bend section of the duct meets the fixed inlet and outlet. On comparing the adjoint sensitivities for the initial (shown in Fig. 14) and the optimised geometry (shown in Fig. 21), it can be seen that the optimisation process has significantly reduced the adjoint sensitivities in the bend regions (which were initially highly sensitive), leading to an optimised design which is close to optimum.

The flow streamlines for the initial and optimised geometries are shown in Fig. 22. It can be seen that the initial flow

with spline curves, (ii) to increase the number of control points used to define a spline, thereby providing additional shape flexibility compared to the line entities they replaced and (iii) to insert new sketch profiles along a multi-section solid such that there is more control over the shape of a

Fig. 19 Evolution of parametric effectiveness during the optimisation

field possesses greater non-uniformity compared to the optimised flow. The adjoint sensitivities suggested a surface movement that would reduce the high adverse pressure gradients observed on the high-curvature areas of the S-bend. As the result of optimisation, the flow requires less energy to remain attached, subsequently reducing the recirculation region, and thus reduces the objective function. This is seen in Fig. 23, where the contours of velocity magnitude are plotted at different cross-sections along the duct.

6 Discussion

In this work, a process has been described where the parameterisation of a CAD model has been modified automatically using the python APIs to the CAD system. This provides the opportunity to achieve a better optimum. The approaches used were to (i) replace line entities in a sketch body along its length. The first two will be beneficial to all sketch-based features in a CAD system (e.g., boss, revolution etc.). Adjoint surface sensitivities have been used to predict the effect of adding features since this allows the effect of many different shape changes to be evaluated using a single analysis. The metric for determining whether a new feature was needed was parametric effectiveness, where a low value indicated a need for improvement. The approach outlined may sometimes result in the features getting inserted where

Table 1 Time taken by the optimisation process

Optimisation	Time (approx.)
Original parameters	14 h
Adaptive parameterisation	22 h
Progressive parameterisation	29 h
Static parameterisation	19 h

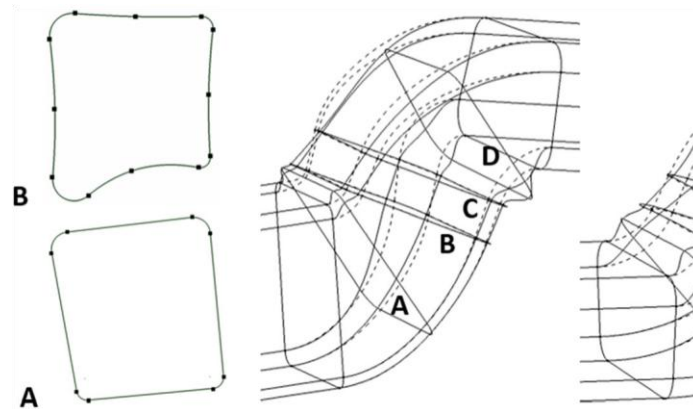
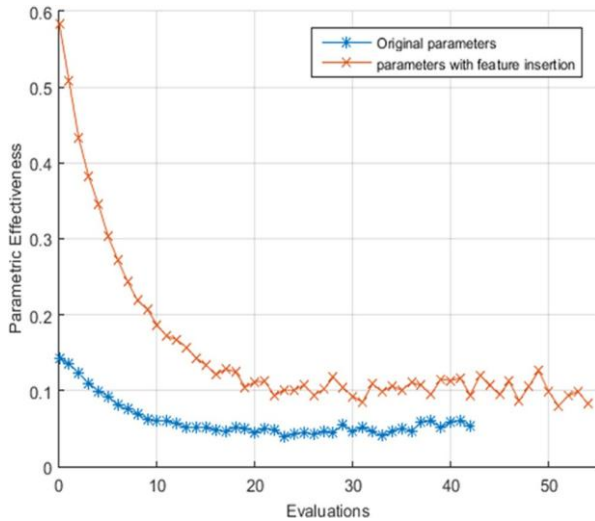


Fig. 20 Comparison between original (broken lines) and optimised (solid line) CAD methodology (middle images from opposite orientations)

parameters already exist or where more design flexibility is not required. But, these situations can be identified by a decrease in the value of parametric effectiveness when such features are inserted. This



happens because as increasing number of parameters are moving the same region of the boundary, or are moving less sensitive regions, the parameterisation will be penalised by the unit root-mean-squared constraint inherited in the calculation of parametric effectiveness. The process has been implemented such that the current state of a model is saved before new features are added. Should adding the new features or parameters not improve the parametric effectiveness then the process will revert to the previous model.

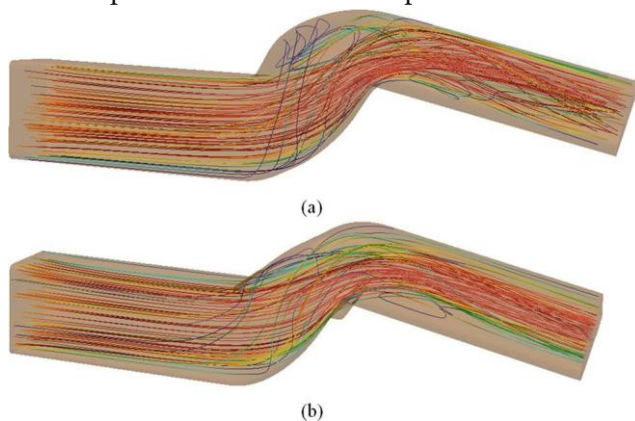


Fig. 22 Flow streamlines for a initial and b optimised geometry

The Python Scipy (Scipy Optimize 2017) optimisation framework used in this work was coupled with the developed tools such that it automatically inserts new CAD features at the start of the optimisation or during the optimisation. One of the difficulties in automated shape optimisation using

CAD is to establish an appropriate lower- and upper-bounds of the design variables. This is usually required to ensure the robustness of the process such that the CAD model boundary topology does not change and

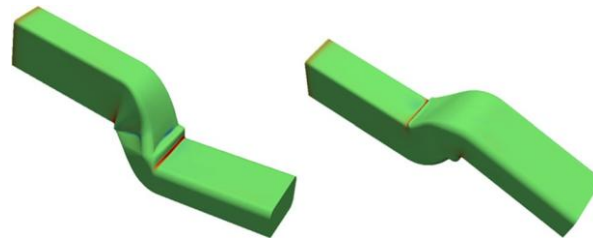
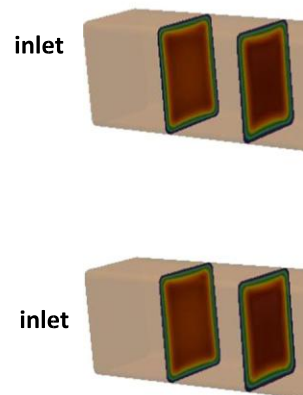


Fig. 21 Adjoint sensitivities contour on the optimised geometry

that the model can regenerate. The approach that is used in this work is robust against boundary topology changes. To ensure the robustness of the CAD model update, the CAD system API is used to perturb each parameter by a fixed amount ($\pm B_{max}$) and ensures the CAD model is updated successfully. If the model fails to update, then the perturbation size is reduced by 10% and reattempted until the model successfully regenerates. The resulting perturbation is used to compute sensitivities for that parameter. The same process is used for the newly inserted control points.

For the beam model, the finite element mesh generation and analysis were not very expensive (in the order of seconds). The insertion of new CAD features was done using an iterative approach, where new

Fig. 23 Contour plots of velocity magnitude for the a initial and b optimised geometry



features were added once the optimisation was complete for the current set. In Section 5, where computationally expensive CFD runs were performed during the optimisation, an iterative approach for generating feature-based CAD models using feedback from CFD simulations (Robinson et al. 2013) became prohibitive. So, the proposed approach is to use the techniques described in this paper to refine the CAD parameterisation upfront, and then optimising the model. The comparison of optimisation results for power-loss across the duct for initial parameters and the ones obtained using the automatic parameterisation scheme for feature insertion is shown in Fig. 18, where it can be seen that the CAD model with highest parametric effectiveness resulted in comparatively higher reduction in objective function compared to the ones with lower parametric effectiveness. Also, it was interesting to find that the optimisation using different strategies converged to different local minima, and the static parameterisation strategy with new features inserted at the start of optimisation resulted in better optimised geometry compared to the optimums reached when insertion of features was done during the optimisation. Further, analysing the computational time required for the two optimisation processes in Table 1, it was found that the optimisation process with newly added features took more time compared to the optimisation using the original parameters. This increased computational time resulted from the increase in number of iterations taken by the optimiser for convergence. Although new features resulted in more than 7% reduction in power-loss compared to that achieved with the original parameters, there is the potential for the computational cost to become a limiting factor. In such cases, either the optimiser should be limited by the maximum number of CFD runs, or other optimisation methods with higher order of convergence can be employed, which is an interesting area for further research.

7 Conclusion

Through this work, it has been shown that:

- & Automated python scripts have been developed to automatically insert CAD features for modifying 2D sketches and multi-section solids built in CATIA V5.
- & The parametric effectiveness of inserting new features to the original CAD model can be calculated using a sin

References

- Agarwal D, Robinson TT, Armstrong CG, Marques S, Vasilopoulos I, Meyer M (2018) Parametric design velocity computation for CADbased design optimization using adjoint methods. *Eng Comput* 34: 225–239. <https://doi.org/10.1007/s00366-017-0534-x>
- Anderson CW, Crawford-Hines S (2000) Fast generation of nurbs surfaces from polygonal mesh models of human anatomy. Colorado State University Computer Science Technical Report CS-99 101
- Ansys ICEM CFD (2017) <http://www.ansys.com/en-GB/Services/training-center/platform/introduction-to-ansys-icem-cfd-Hexa>. Accessed 09/27/2017
- Auriemma S, Banovic M, Mykhaskiv O, Legrand H, Müller J, Walther A (2016) Optimisation of a U-bend using CAD-based adjoint method with differentiated CAD kernel, VII European Congress on Computational Methods in Applied Sciences and Engineering, <https://www.eccomas2016.org/proceedings/pdf/10065.pdf> Accessed 02/18/2018
- Banović M, Mykhaskiv O, Auriemma S, Walther A, Legrand H, Müller JD (2018) Algorithmic differentiation of the Open CASCADE Technology CAD kernel and its coupling with an adjoint CFD solver, *Optimization Methods and Software*, 33:4-6, 813-828, <https://doi.org/10.1080/10556788.2018.1431235>
- Belegundu A, Rajan S (1988) A shape optimization approach based on natural design variables and shape functions. *Comput Methods Appl Mech Eng* 66:87–106. [https://doi.org/10.1016/0045-7825\(88\)90061-8](https://doi.org/10.1016/0045-7825(88)90061-8)
- Burgos MJ (2015) NURBS-based geometry parameterization for aerodynamic shape optimization. Dissertation, PhD thesis, Universidad Politecnica de Madrid
- CATIA V5 (2017) <http://www.3ds.com/products-services/catia/>. Accessed 01/27/2017
- Chen S, Tortorelli D (1997) Three-dimensional shape optimization with variational geometry. *Struct Optim* 13:81–94. <https://doi.org/10.1007/BF01199226>
- Geuzaine C, Remacle JF (2009) GMSH: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *Int J Numer Methods Eng* 79:1309–1331. <https://doi.org/10.1002/nme.2579>
- Giles MB, Pierce N (2000) An introduction to the adjoint approach to design. *Flow Turbul Combust* 65:393–415. <https://doi.org/10.1023/A:1011430410075>
- Guo B (1997) Surface reconstruction: from points to splines. *Comput Aided Des* 29:269–277. [https://doi.org/10.1016/S0010-4485\(96\)00055-3](https://doi.org/10.1016/S0010-4485(96)00055-3)
- Hammond M, Robinson A (2000) Python programming on win32: Help for windows programmers. O'Reilly Media Inc
- Hardee E, Chang K, Tu J, Choi KK, Grindeanu I, Yu X (1999) A CADbased design parameterization for shape optimization of elastic solids. *Adv Eng Softw* 30:185–199. [https://doi.org/10.1016/S0965-9978\(98\)00065-9](https://doi.org/10.1016/S0965-9978(98)00065-9)
- Helgason E, Krajnovic S (2012) Aerodynamic shape optimization of a pipe using the adjoint method. *ASME International Mechanical Engineering Congress & Exposition* 7:9–15. <https://doi.org/10.1115/IMECE2012-89396>

- Hoogervorst JE, Elham A (2017) Wing aerostructural optimization using the individual discipline feasible architecture. *Aerosp Sci Technol* 65:90–99. <https://doi.org/10.1016/j.ast.2017.02.012>
- Jameson A (2004) Efficient aerodynamic shape optimization. 10th AIAA/ISSMO multidisciplinary analysis and optimization conference, multidisciplinary analysis optimization conferences. <https://doi.org/10.2514/6.2004-4369>
- Jameson A, Martinelli L, Pierce N (1998) Optimum aerodynamic design using the Navier–Stokes equations. *Theor Comput Fluid Dyn* 10: 213–237. <https://doi.org/10.1007/s001620050060>
- Jesudasan R, Zhang X, Müller J (2017) Adjoint Optimisation of Internal Turbine Cooling Channel Using NURBS-Based Automatic and Adaptive Parametrisation Method. ASME. Gas Turbine India Conference, Volume 1: Compressors, Fans and Pumps; Turbines; Heat Transfer; Combustion, Fuels and Emissions. <https://doi.org/10.1115/GTINDIA2017-4669>
- Karpouzas GK, Papoutsis-Kiachagias EM, Schumacher T, Villiers Ed, Giannakoglou KC, Othmer C (2016) Adjoint optimization for vehicle external aerodynamics. *Intl J Automot Eng* 7:1–7. https://doi.org/10.20485/jsaeijae.7.1_1
- Kenway G, Kennedy G, Martins J (2010) A CAD-free approach to highfidelity aerostructural optimization. 13th AIAA/ISSMO multidisciplinary analysis optimization Conference, Conference, Multidisciplinary Analysis Optimization Conferences <https://doi.org/10.2514/6.2010-9231>
- Kripac J (1997) A mechanism for persistently naming topological entities in history based parametric solid models. *Computer-Aided design* 29(2): 113–122. [https://doi.org/10.1016/S0010-4485\(96\)00040-1](https://doi.org/10.1016/S0010-4485(96)00040-1)
- Latif M, Boyd R, Hannam R (1993) Integrating CAD and manufacturing intelligence through features and objects. *Int J Comput Integr Manuf* 6:87–93. <https://doi.org/10.1080/09511929308944558>
- Mader CA, RA Martins J, Alonso JJ, Der Weide EV (2008) Adjoint: an approach for the rapid development of discrete adjoint solvers. *AIAA J* 46:863–873. <https://doi.org/10.2514/1.29123>
- Martin MJ, Andrés E, Widhalm M, Bitrián P, Lozano C (2012) Nonuniform rational B-splines-based aerodynamic shape design optimization with the DLR TAU code. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 226:1225–1242. <https://doi.org/10.1177/2F0954410011421704>
- Mei Y, Wang X, Cheng G (2008) A feature-based topological optimization for structure design. *Adv Eng Softw* 39:71–87. <https://doi.org/10.1016/j.advengsoft.2007.01.023>
- Nadarajah S, Jameson A (2001) Studies of the continuous and discrete adjoint approaches to viscous automatic aerodynamic shape optimization, 15th AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-located Conferences, <https://doi.org/10.2514/6.2001-2530>
- Othmer C (2008) A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows. *Int J Numer Methods Fluids* 58:861–877. <https://doi.org/10.1002/flid.1770>
- Papoutsis-Kiachagias E, Giannakoglou KC (2016) Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: industrial applications. *Arch Comput Methods Eng* 23:255–299. <https://doi.org/10.1007/s11831-014-9141-9>
- Python 3.5 (2017) <http://www.python.org/downloads/release/python350/>. Accessed 10/05/2017
- Robinson TT, Armstrong CG, Chua HS (2012) Strategies for adding features to CAD models in order to optimize performance. *Struct Multidiscip Optim* 46:415–424. <https://doi.org/10.1007/s00158012-0770-z>
- Robinson TT, Armstrong CG, Chua HS (2013) Determining the parametric effectiveness of a CAD model. *Eng Comput* 29:111–126. <https://doi.org/10.1007/s00366-011-0248-4>
- Salomons OW, van Houten FJ, Kals H (1993) Review of research in feature-based design. *J Manuf Syst* 12:113–132. [https://doi.org/10.1016/0278-6125\(93\)90012-1](https://doi.org/10.1016/0278-6125(93)90012-1)
- Samareh J (2001) Survey of shape parametrisation techniques for highfidelity multidisciplinary shape optimization. *AIAA J* 39:877–884. <https://doi.org/10.2514/2.1391>
- Samareh J (2004) Aerodynamic shape optimization based on free-form deformation. 10th AIAA/ISSMO multidisciplinary analysis and optimization conference, multidisciplinary analysis optimization conferences. <https://doi.org/10.2514/6.2004-4630>
- Scipy Optimize (2017) <https://docs.scipy.org/doc/scipy/reference/optimize.html>. Accessed 01/27/2017
- Sederberg TW, Parry SR (1986) Free-form deformation of solid geometric models. In Proceedings of the 13th annual conference on Computer graphics and interactive techniques ACM 20: 151–160. <https://doi.org/10.1145/15922.15903>
- Shah JJ (1991) Assessment of features technology. *Comput Aided Des* 23:331–343. [https://doi.org/10.1016/0010-4485\(91\)90027-T](https://doi.org/10.1016/0010-4485(91)90027-T)
- TAPENADE (2018). <https://www-sop.inria.fr/tropics/tapenade.html>. Accessed 02/21/2018
- Torreguitart IS, Verstraete T, Mueller L (2017) Optimization of the LS89 axial turbine profile using a cad and adjoint based approach. *International Journal of Turbomachinery, Propulsion and Power* 3: 20. <https://doi.org/10.3390/ijtpp3030020>
- Walther A, Griewank A (2010) A Package for the Automatic Differentiation of Algorithms Written in C/C. URL <https://projects.coin-or.org/ADOL-C> Accessed 02/18/2018
- Wang MY, Wang X, Guo D (2003) A level set method for structural topology optimization. *Comput Methods Appl Mech Eng* 192: 227–246. [https://doi.org/10.1016/S0045-7825\(02\)00559-5](https://doi.org/10.1016/S0045-7825(02)00559-5)
- Xu S, Jahn W, Müller J (2014) CAD based shape optimization with CFD using a discrete adjoint. *Int J Numer Methods Fluids* 74:153–168. <https://doi.org/10.1002/flid.3844>
- Xu S, Timme S, Mykhaskiv O, Müller J (2017) Wing-body junction optimisation with CAD-based parametrisation

including a moving intersection. *Aerosp Sci Technol* 68:543–551. <https://doi.org/10.1016/j.ast.2017.06.014>

Yu G, Müller J, Jones D (2011) CAD based shape optimization using adjoint sensitivities. *Comput Fluids* 46:512–516. <https://doi.org/10.1016/j.compfluid.2011.01.043>