Digital Geometry to Support a Gas Turbine Digital Twin

Bill Dawes [1]
Whittle Lab., Cambridge University Engineering Department, Cambridge UK

Nabil Meah, Andrey Kudryavtsev, Rich Evans, Matt Hunt, Phil Tiller
Cambridge Flow Solutions Ltd, The Bradfield Centre, Cambridge Science Park, Cambridge CB4 0GA, UK

"Digitalisation", Industry 4.0, the Digital Twin, Data-driven design & manufacturing are set to revolutionise the way we do business. This paper considers this in the engineering world of power generation, gas turbines & power plant - and asks the question: What will it take to make a Digital Twin real. A Digital Twin must have physics-based simulation at its heart and confront three major challenges: Scale of simulation; Scaling the simulation; Responding to data-driven feedback. This paper will discuss these issues in turn and make the case that the ability to represent & manage geometry is the Digital Thread which supports a Digital Twin. We discuss the use of classical BREP CAD and the new Digital Geometry solid modelling kernel we have been developing. We illustrate with examples of recent work we have been performing aimed at addressing these challenges.

I. Introduction: what is a Gas Turbine Digital Twin

The motivation for a Digital Twin is derived from several, key business drivers. First of all, the need to bring new products to market faster means continued pressure to replace expensive physical testing with simulation. Next, the need to avoid – or, if necessary, manage – expensive in-service issues will require increased use of simulation (combined with on-site condition monitoring). The inexorable trend is for component-based optimisation to be replaced by system or sub-system optimisation requiring multi-disciplinary simulation of larger assemblies of geometry – and including transients and FSI (flutter etc.). We must think more broadly and use physics-based simulation (defined as conjugate CFD/FEA on systems together with manufacturing simulations, casting, costing, wear & degradation etc…) not just for the functional behaviour of a product but its economic, business behaviour. A successful engineering company in the 21st century will be one that harnesses the benefits of computer-based simulation.

"Digitalisation", Industry 4.0, the Digital Twin, Data-driven design & manufacturing are set to revolutionise the way we do business (Forbes [2017]). A Digital Twin is a virtual model of a process, product or service to accelerate design, develop new opportunities and plan for the future by using simulation. This merging of the digital and physical worlds, the Industrie 4.0 paradigm, is driven by Data collected at all stages over the life of a product, from concept to design to manufacture to service & monitoring of systems and on to retirement. All this Data fed back into the product development cycle. This Data will be analysed using Artificial Intelligence, AI, and will likely be distributed across parallel computer resource, probably the Cloud.

Figure 1 sketches a likely Gas Turbine Digital Twin ranging from temperature-entropy (T-s) cycle diagrams & blade velocity triangles through fully-featured, 3D design, manufacturing, service and Maintenance, Repair & Overhaul (MRO). There are a number of opportunities for a Digital Twin: "*as-designed*" vs "*as-manufactured*"; *remote monitoring & abnormality detection*; *MRO, Maintenance, Repair & Overhaul.*

[1] AIAA Senior Member

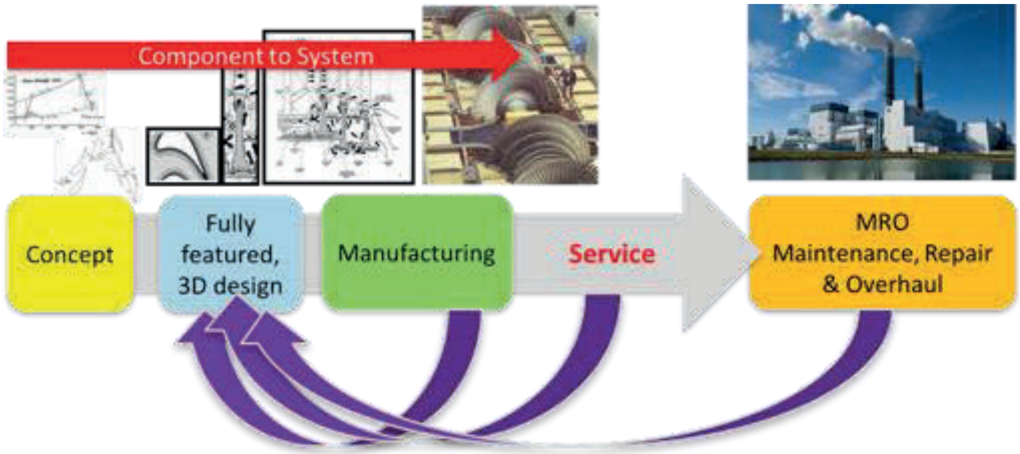

Fig.1: A Digital Twin for a Gas Turbine power plant with data-driven feedback from manufacturing, service and MRO

The first opportunity for a Digital Twin is: "*as-designed*" vs "*as-manufactured*". There will be discrepancies between the design intent, as expressed in the Manufacturing CAD ("MCAD") due to: the manufacturing process; or assembly; or even the hot-cold-running transformation approximations. For example, Figure 2 shows the effect of manufacturing variability on the performance of HPT blades (Lee *et al* [2018]). *SO*, does this matter? Is a re-design or re-assembly required? This can be a very expensive decision…

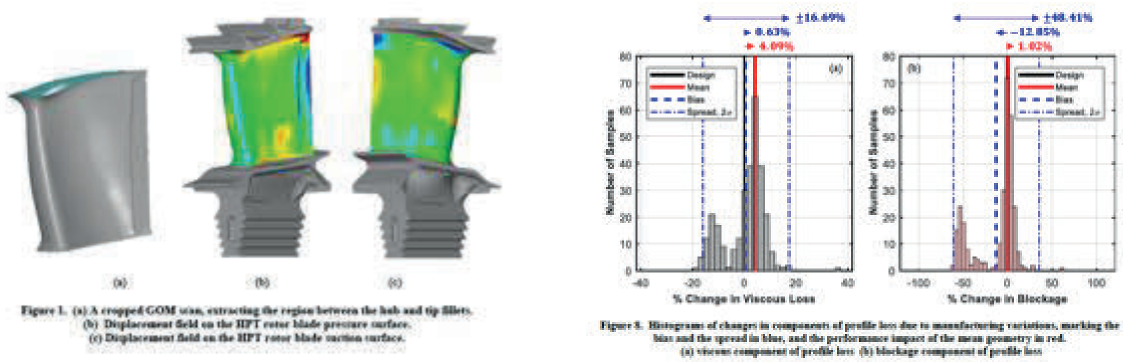

Fig.2 Effect of manufacturing variability on the performance of HPT blades (Lee *et al* [2018])

The second opportunity for a Digital Twin is: *remote monitoring & abnormality detection*. In an installed Gas Turbine power plant, data is measured in real-time: fuel flow, some pressures & temperatures, rpm, vibration, etc…perhaps clearances but no geometry. There may be an abnormality or problem – with various potential causes - each with associated mitigations…Figure 3 shows typical in-service condition monitoring (from Meher-Homji *et al* [2002]) showing the changes in compressor efficiency and Heat Rate in a large gas turbine and a "fouling event"; (right) the Mahalanobis Distance (see *https://blogs.sas.com/content/iml/2012/02/15/what-is-mahalanobis-distance.html*) is used to judge whether a data departure is within the expected range or abnormal. *SO*, maybe use a Digital Twin run with many potential scenarios to generate a training Data Base for an AI (Artificial Intelligence) system which can then manage & interpret the real field data.

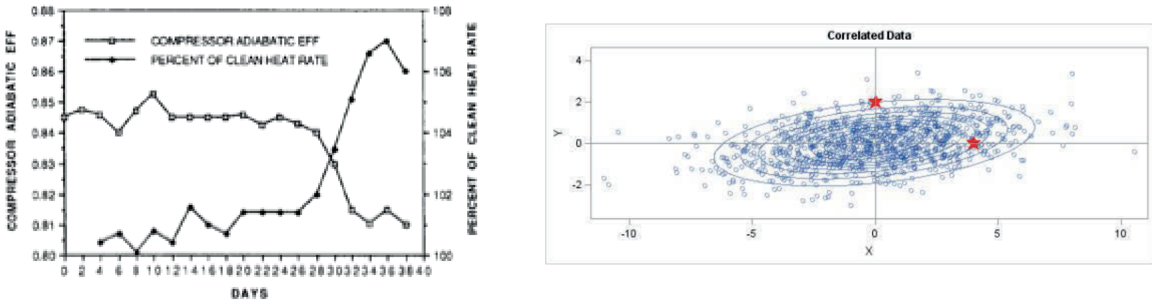

Fig.3 Typical in-service condition monitoring (from Meher-Homji *et al* [2002]) showing the changes in compressor efficiency and Heat Rate in a large gas turbine and a "fouling event"; (right) the Mahalanobis Distance (see *https://blogs.sas.com/content/iml/2012/02/15/what-is-mahalanobis-distance.html*) is used to judge whether a data departure is within the expected range or abnormal

The third opportunity for a Digital Twin is: *MRO, Maintenance, Repair & Overhaul*. Usually life is prioritised over performance (except in very competitive markets) as this delivers a higher return on investment for the operator. *SO*, using the actual observed state of the geometry, what needs to replaced? What still has useful life? What performance penalties might be expected? Figure 4 includes a gas turbine blade coated with volcanic ash deposits (from Dunn *et al* [1994]) – does it need to be removed and repaired or replaced?

Fig.4. Typical in-service MRO issue - volcanic ash deposits on a HPT (from Dunn *et al* [1994]) – do the blades need to be removed and repaired or replaced?

However, despite the opportunity, there are a number of challenges to make a Digital Twin real and these are discussed in the following Sections:

- Scale of simulation;
- Scaling the simulation;
- Responding to data-driven feedback.

II. Challenges to make a Digital Twin real

A. Scale of Simulation

A Digital Twin will need to be able to represent & manage huge geometries over scales ranging from individual components like gas turbine blades through to assemblies and on to systems at the level of complete power plant. The next Figure, Fig.5, shows an application example, a turbocharger installed under-hood in a car. The task is to develop

& improve a whole product, not just separate components individually – a systems-based approach. To set a sense of scale, overall simulation requirements are likely:

- *Very large meshes*: complete engine bay + component CHT mesh + external domain will be +1Bn cells or so
- "*Geometry editing*" = very quick and substantial geometry change, without affecting simulation process speed (e.g. "cut-paste" a complete new manifold design, maybe in real time)
- *Integrated tools* for simulation, not simply assembly of existing packages via "loose-coupling"
- *Data handling* (full parallelism) and visualisation to match size & ambition of simulation.

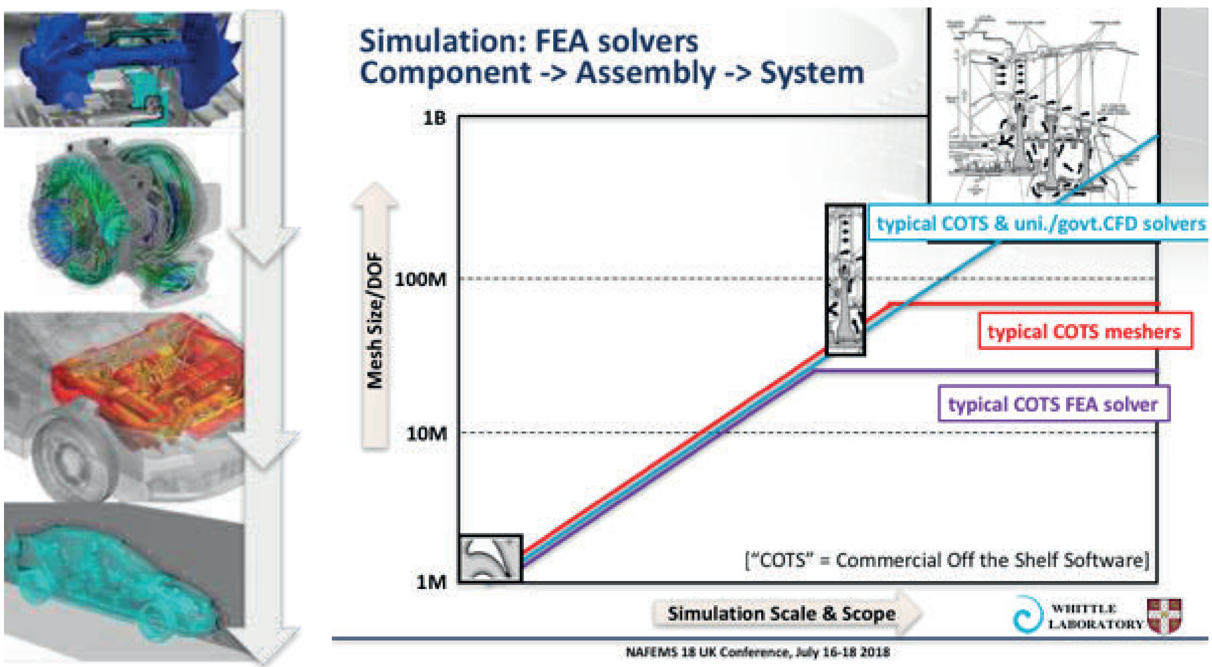

Fig.5: Trends in simulation from component to system-level design drive growing simulation scale and scope

This trend from component to system-level design drives growing simulation scale and scope as illustrated. In terms of mesh size, typical CFD solvers are in good shape and handle very large meshes via efficient parallel implementations. However, the CAD geometry kernel, mesh generation itself and (especially) FEA analyses are severely limited by their serial implementation effectively blocking useful application much beyond sub-assembly scale. Moreover, these are not the only bottlenecks inhibiting application of conventional simulation systems to a Digital Twin – the workflow itself must be considered.

B. Scaling the simulation

A Digital Twin is at core an integrated physics-based workflow combining a range of multi-disciplinary tools. A Digital Twin will need to be distributed across parallel computer resource, perhaps the Cloud, supporting huge, conjugate, aero-thermal-mechanical simulations on huge meshes. When the simulation scale becomes large, the amount of memory RAM to support the simulation becomes large – so, many computer cores are needed. With typical consumption of around 1GB per 1M mesh cells (for real-world practical unstructured meshes) a simulation of 1B cells needs 1000 cores – it is a simple as that. *The benefit of an HPC system is more in absolute job size rather than job speed.*

The key to exploiting the benefit offered by an HPC system is to understand *scaling* - for best efficiency care must be taken for the *compute load* to be much bigger than *the communicate load* - eventually as the number of cores increases *communication* dominates *compute* load and the speed up asymptotes as sketched in Figure 6. In addition, *Amdahl's Law* [1967] makes clear that any *serial* parts of the code will eventually dominate – and very savagely. For example, if 10% of the code is serial then the maximum speed up on any HPC is no more than factor 10 ! (The run time of the 90% parallel code is driven towards zero on a large core count machine but the residual serial code stays serial…) In the context of a Digital Twin, the key observation is that Amdahl's law applies to the *overall workflow* consisting of CAD-to-mesh-to-simulation-to-visualisation-to-geometry edit etc. as illustrated below.

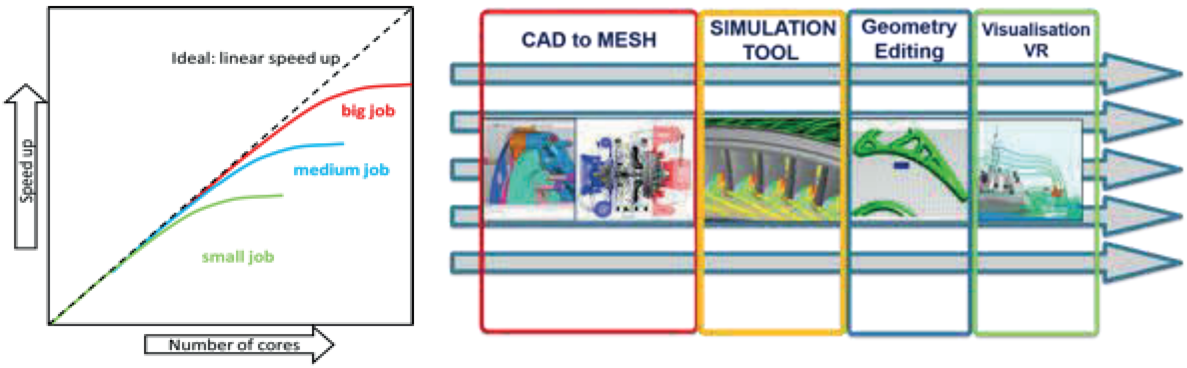

Fig.6: Scaling an integrated process chain: Amdahl's Law

Accordingly, a useful *Simulation Environment* to support an integrated workflow, a Digital Twin, must be implemented *end-to-end* in parallel including the geometry kernel, mesh generator, *close-coupled* with simulation & post-processing combined with geometry editing & management to enable scripted, automated design and data-driven feedback from the through-life behaviour of the product. This observation is closely aligned with the NASA Vision 2030 study on the future of simulation & HPC (Slotnik *et al* [2014]).

C. Responding to data-driven feedback

The third, and perhaps most significant, challenge to be confronted to allow the creation of a useful Digital Twin is the need to respond in a physics-based way to Data-driven feedback derived from field data and in-service monitoring. Knowledge about wear, damage, degradation, etc. will not only need to be incorporated "upstream" into improved design & manufacturing but also "through-life" into assessing & managing, in a financially quantitative way, ongoing MRO (Maintenance, Repair & Overhaul) needs. This can only done by responding, in the most general way, to *geometry*.

Fig.7: Typical in-service degradation; a meaningful Digital Twin must be able to respond to this "Data-driven" feedback; (left) hot corrosion (Wing *at al* [1981]), (centre) deposition & erosion on a HPT NGV (Batalha [2012]); (right) rendered laser scan of a volcanic ash damaged HPT NGV (Bonilla [2012])

To illustrate the scale of the challenge, Figure 7 shows typical in-service degradation in aero-engines. In the Figure on the left is hot corrosion (Wing *at al* [1981]), in the centre, deposition & erosion on a HPT NGV (Batalha [2012]) and on the right a rendered laser scan of a volcanic ash damaged HPT NGV (Bonilla [2012]). The particular HPT NGV shown in the centre had apparently been under observation via borescope and, despite its appearance, at the next

Shop Visit was declared within limits and sent for repair rather than being scrapped (the replacement cost is around $10-15k per blade (Batalha [2012])).

The repair-replace judgement – as well as the judgement that the blade was safe to continue flying – were made using experience and naturally in a conservative way. Aero-engine maintenance is very expensive: the typical cost of "performance recovery" at a Shop Visit is $1.5-2M per engine and the replacement cost of the several Life Limited Parts (HPT module, etc.) is in the range $400-600k each (Ackert [2011]). Given the extremely large costs involved and the current trend to bundle maintenance costs into a long term support package, there is huge scope for a much more physics-based approach.

A meaningful Digital Twin, therefore, must be able to respond to this "Data-driven" feedback. Knowledge of damage, degradation, etc. will need to be incorporated into improved design & manufacturing and into in-service monitoring & management. The only way forward is to replace judgement with physics-based simulation. *And this can only be done via the geometry.*

III. Geometry is the Digital Thread

It our firm contention that a Digital Twin for a complete GT power plant must be based on *physics-based simulation* and will need to be able to represent & manage huge geometries over scales ranging from individual *components* like gas turbine blades through to *assemblies* and on to *systems* at the level of complete power plant. The Digital Twin must encompass through-life *economics modelling* including MRO return for OEM & Operator ROI and the ability to respond to *environmental & regulatory* demands.

The key question is not the idea idea itself but rather:

- *what would it take to make a Digital Twin real* – what is the *Digital Thread*?

The standard response to the change of geometry due to wear or degradation is to morph the associated simulation *mesh* – see for example the work on ash deposition in turbine cooling holes reported by Forsyth *et al* [2017] or deposition on turbine blades reported by Casari *et al* [2017]. BUT to perform a component life analysis needs a conjugate, coupled aero-thermal-mechanical analysis – this needs a modified *geometry*.

Our contention is that *Geometry* is the Digital Thread. The role of *Geometry* is to support simulation; geometry must be available throughout the simulation process chain, in a form suited to CFD, FEA, etc… and be capable of being data-driven and so respond to manufacturing variability, in-service issues, etc…SO, the next question is: what sort of "geometry"?

"Geometry" is generally thought of as classical NURBs-based BREP CAD. This approach is very widely used and embedded within industrial design and PLM systems but has two key drawbacks relative to the needs of the Digital Twin. Firstly, the approach is basically serial (in the sense it cannot be distributed over parallel compute cores); secondly, it responds inefficiently & slowly (and via human intervention if any inbuilt parameterisation cannot cope) to "data-driven feedback". In short, BREP-NURBS is not created with enough degrees of freedom to accurately capture & reflect in-service, degraded geometry. Finally, in general, classical CAD is still not well aligned with the needs of automated mesh generation and hence not well suited to automated simulation workflows (see for example Gammon *et al* [2018]).

In terms of parallelism, Haimes & Dannenhoffer [2018], with their work on EGADSlite and EGADS/OpenCASCADE, have shown how an scalable, parallelisable geometry kernel can be constructed – but still based on classical trimmed BREP concepts. This potentially solves the scalability challenge, but not the ability to respond to data-driven feedback. We have been developing an alternate approach (Dawes *et al* [2018]) based on *Digital Geometry* which overcomes both of the drawbacks of standard BREP CAD; this will be described in the following Section.

IV. Digital Geometry

The famous Bresenham line algorithm (1962) was developed as a way of representing a line via discrete *pixels* – "rasterisation" on the newly emerging Cathode Ray Tube terminals. This is essentially the core idea in digital photography – a picture – in 3D this becomes *geometry* and pixels, *voxels*. Our *Boxer* software (see Dawes *et al* [2005-2018]) is built on Digital Geometry using generalised 3D versions of the fundamental Bresenham algorithm; Figure 8 illustrates this. This consists of an integer representation of geometry down to a chosen length scale – voxels which determine "spatial occupancy": either occupied, vacant or cut. This is combined with a local Distance Field managed through Level-Set technology – to represent sub-voxel scale geometry. This "distance" is literally & simply the physical distance from the cell centres (or vertices) to the closest geometry and is just a scalar field variable, just like fluid density or pressure.

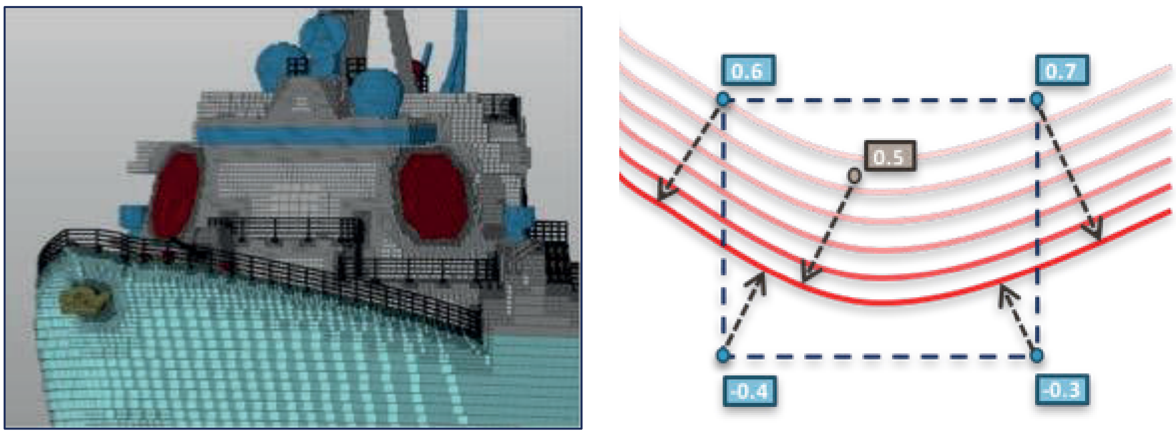

Fig.8: The Digital Geometry Kernel in *Boxer*; on the left the 3D voxel image; on the right the Distance Field storing sub-voxel scale geometry information

There are two key advantages of this approach:

- Digital Geometry can be distributed onto any cluster - enables true parallel scalability
- Geometry editing & management is supported in a very general, topology-indpendent way

Looking ahead at the Simulation System of the Future, geometry will need to be available throughout the process chain to support solution adaptive mesh refinement, Fluid Structure Interaction, and automated design optimization. The simulation sizes will be in the Billions of mesh cells, supporting conjugate analysis, and the process chain will have to be *end-to-end parallel* with no serial bottlenecks. Hence the geometry modeling itself must be capable of being implemented & scaling in parallel – this is trivial for our Digital Geometry kernel but very difficult to imagine with a kernel based on traditional NURBS-BREP constructs.

In the Digital Geometry world, editing & managing the geometry consists of modifying the Distance Field as illustrated in the Figure 9: shown on the left, simple Boolean operations; on the right morphing of one shape, "A", to another, "B".

In more detail, the Level-set Distance Field is defined by: $\phi(x,\, t > 0) \;=\; \pm d$. It is easy to show (Adalsteinsson *et al* [1995]) that there is an associated evolution equation :

$$\phi_t + F \cdot |\nabla\phi| = 0$$

where F is the *speed function* – setting the rate (and also the extent & location) of changes to the Digital Geometry surface $\phi=0$. "Geometry edits" are just changes to this scalar field, defined somehow/anyhow via the function F.

Fig.9: Digital Geometry editing the Distance Field: (left) simple Boolean operations; (right) morphing Shape "A" to Shape "B"

For example:

- standard Boolean operation is just F=min(target_shape, tool_shape)
- classical morphing operation (Breen & Whitacker [2001]) is just $F = (\phi_{target} - \phi_{source})$
- erosion/corrosion/burning/ablation is just F=volume(mass) transfer to/from surface

This geometry framework provides a huge and as yet largely unexplored capability.

As a very simple illustration, Figure 10 shows stages in the simulation of erosion by sand of the surface of a sphere. The erosion simulation was performed using standard Fluent™ Discrete Particle Modelling (DPM) (see [www.ansys.com]) combined with classical erosion modelling (see for example Finnie *et al* [1992]). The erosion model takes the form:

$$ER = \sum_{trajectories} \frac{m_p C(d_p) f(\alpha) v_p^n}{A_{face}}$$

where the erosion rate (ER) of the geometry face depends on the mass of impacting particles, an empirical function of the particle diameter, an empirical function of the impact angle and the particle velocity (with an empirical exponent). The predicted surface variation in erosion rate (m^3/m^2 s) is related *directly* to the Level-set morph Speed Function "F" (m/s) and is imported *directly* into *Boxer* and used to morph the Level Set Distance Field; in the Figure the zero distance (ϕ=0) is rendered.

Fig.10: Stages in the erosion of a sphere subject to sand particle impingement; the image shows the zero distance of the Level Set geometry

A *new* simulation mesh is then automatically generated for this *new*, morphed geometry – see Figure 11. This is a key novelty of our approach – we morph the geometry *not* the mesh.

Boxer is very robust and to generate a mesh for a geometry like this is straightforward; the exported mesh then drives, for example, the Fluent[TM] flow solver allowing the performance of the modified geometry to be *predicted*. Thereby, a scripted, automated "*morph-mesh-solve*" workflow can be created as sketched in Figure 12; it is this workflow which we believe is capable of supporting through-life simulations by coupling physics-based geometry degradation to predicted performance changes.

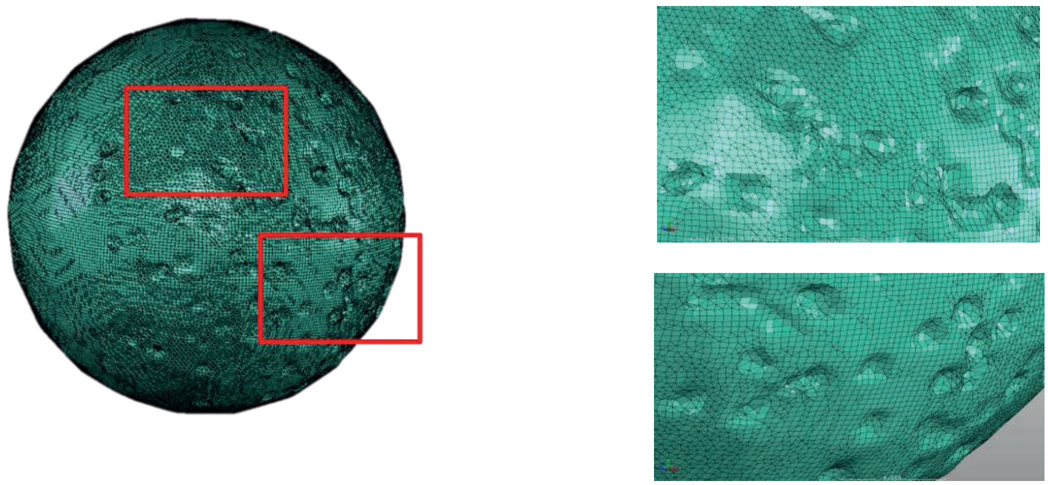

Fig.11: Simulation mesh (surface shown) generated by *Boxer* for the eroded sphere generated from the morphed Level Set geometry

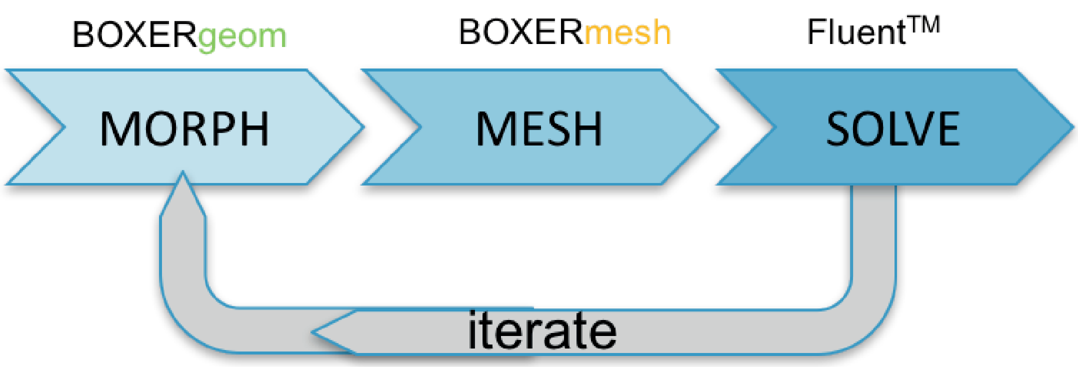

Fig.12: The automated *morph-mesh-solve* workflow

V. Example applications

A. Turbine blade in-service degradation; simulated erosion

This Section will show our first attempts at modelling blade degradation using physics-based simulation. We have chosen particulate erosion as this is one of the most severe mechanisms as discussed earlier. There are a number of approaches to particulate erosion modelling in the literature based on "dilute phase" (one-way coupling) Lagrangian particle tracking combined with an impact "bounce" or "stick" erosion model. As validation, there are a number of published cases: for example Tabakoff *et al* [1990], Hamed, Tabakoff *et al* [2005] or Graham *et al* [2009].

To illustrate typical capability, Figure 13 shows measured erosion by sand particles of an aluminium cylinder/endwall compared with a Fluent™ simulation (taken from Graham *et al* [2009]). The predicted surface erosion rate matches well with the observed erosion pattern. Accordingly, we use in this paper the basic DPM/erosion model available in Fluent™ (based on Finnie *et al* [1992]) and we have reproduced the results in Figure 13 to verify our approach. We have studied both the stator vanes and the downstream rotor blades in a stage; results are presented in the following Section for the vanes.

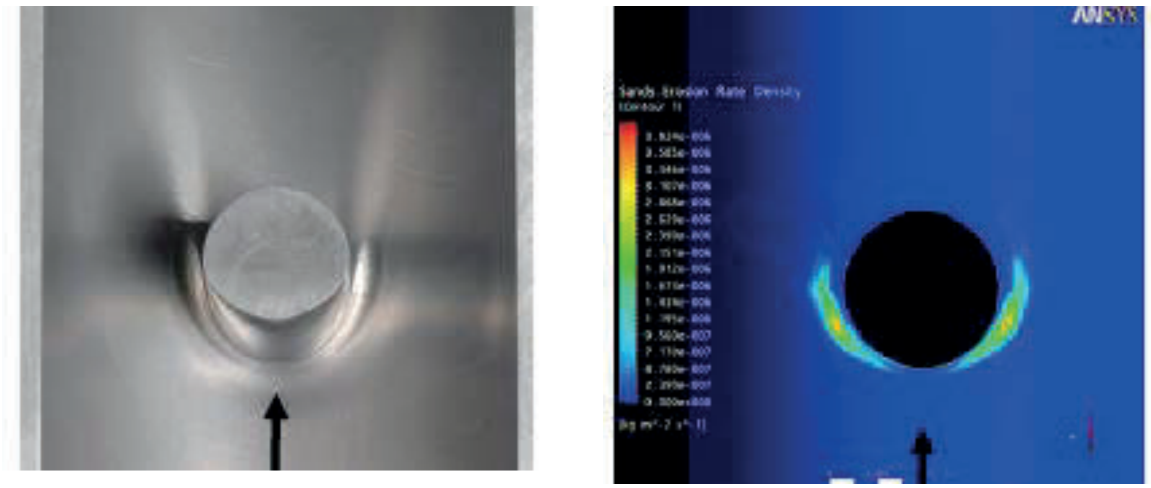

Fig.13: Measured erosion by sand particles of an aluminium cylinder/endwall compared with a Fluent™ simulation of surface erosion rate (taken from Graham *et al* [2009])

The stator blades are typical of those used in an aero-engine HPT stage. A standard unstructured mesh was generated by BOXERmesh and then exported to Fluent™ to produce the baseline solution with standard inlet & outlet boundary conditions. Then the DPM/erosion simulation was performed by Fluent™ injecting sand particles with a "Rosin-Rammler" distribution and specifying the sand mass flow rate / number of particles & the distribution of particle diameters as shown in the Table. The particles were injected in the same direction and with the same velocity as the bulk air flow. Quantitative data on particulates in aero-engine turbines is rarely available in the public domain; for this study the particle size & burden were selected based on the information in Hamed, Tabakoff *et al* [2005] and is representative of "runway sand".

Blade inlet angle [°C]	0
Blade exit angle [°C]	70
Re [-]	0.5-1x10^6
Sand/air mass flow ratio [%]	0.21
Sand: min; mean; max diam.[μm]	60; 250; 1000

Table: Basic stator vane simulation parameters

Different particle sizes take different paths through the flow with small ones following the flow and larger ones having a more "ballistic" trajectory. Figure 14 illustrates this. An overview of the predicted surface erosion rates is shown in Figure 15. The particles in the simulation input are relatively large and their trajectories tend towards "ballistic" so the majority of the particle-blade impacts occur on the blade pressure side.

Notable in Figure 15 is the tendency for particles to collect in the pressure side/endwall corner and for their pathlines to roll up into a vortical structure (see zoom view in Figure 15). This leads to greatly increased local erosion rates compared to the general blade background level of around 0.5mm/10,000h; the predicted maximum erosion rate is around 2.3mm/10,000h. (10,000h is about half the Flight Hours expected before a Restoration Shop Visit.)

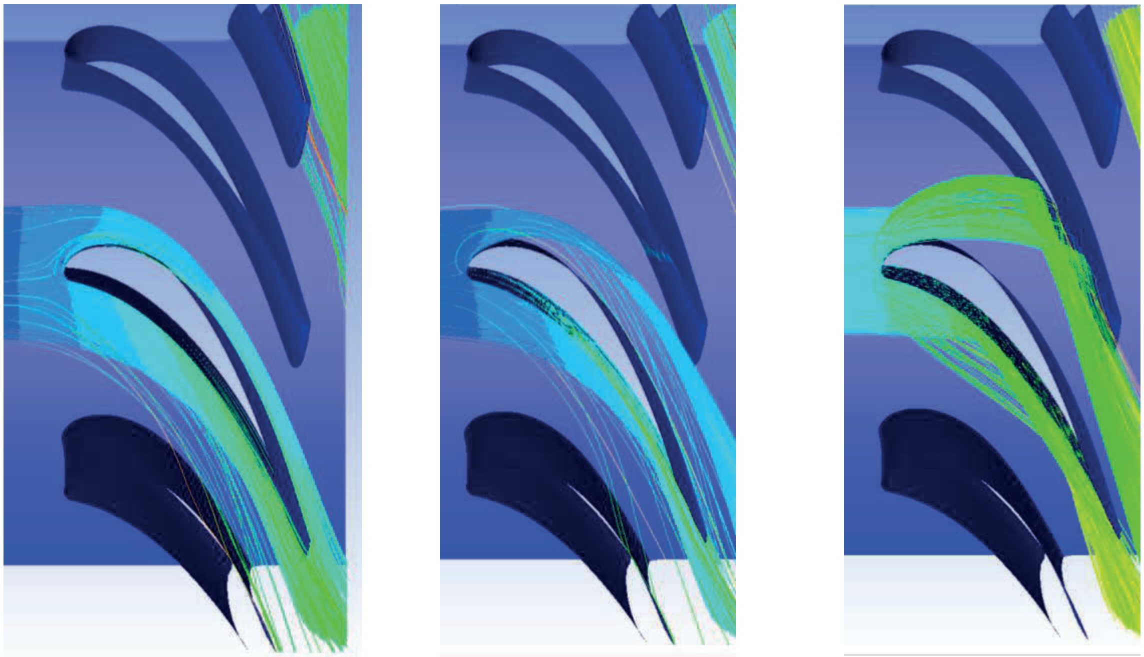

Fig.14: Predicted trajectories of different sizes of particle: (left) 2 µm; (centre) 20 µm; (right) 200 µm

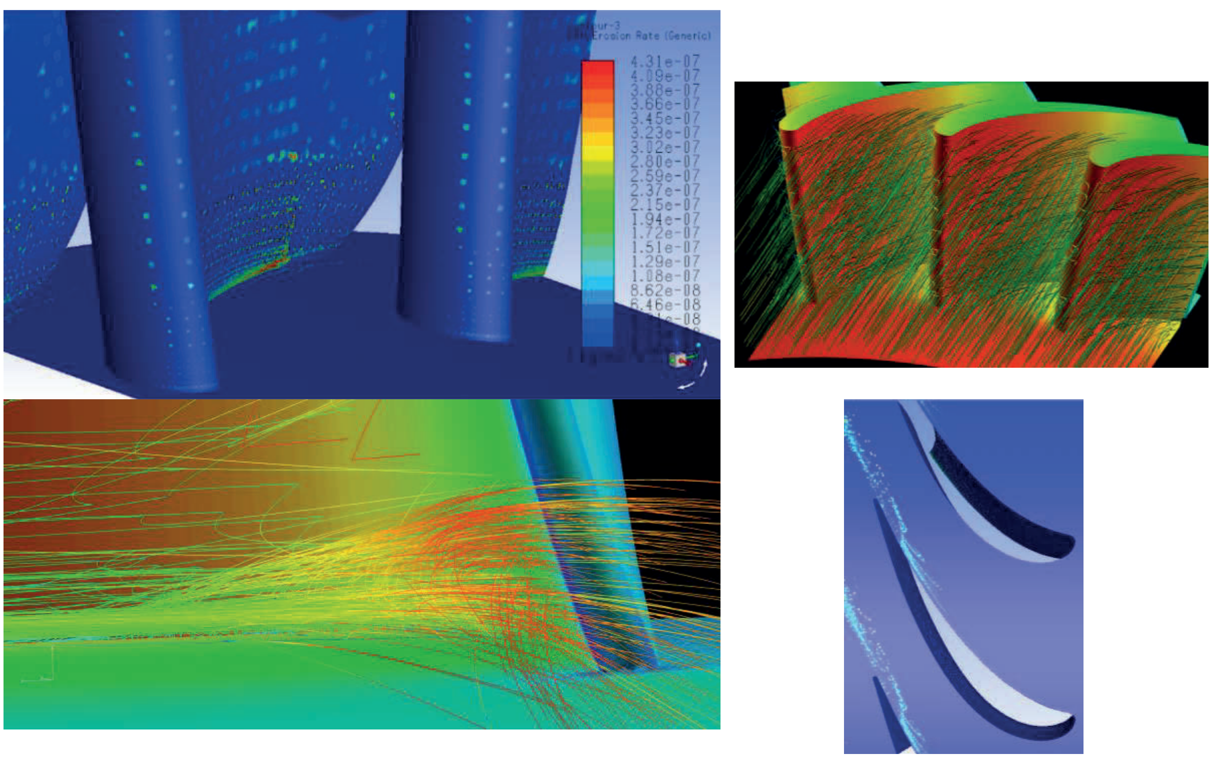

Fig.15: Overview of predicted erosion rates in kg/m^2 s for the HPT stator vane; (zoom view) particle trajectories in the pressure side/endwall corner showing a vortical structure (view upstream from the blade trailing edge)

The next step in the *morph-mesh-solve* workflow is for the erosion rate to be imported into BOXERgeom and for the blade geometry to be morphed using the Speed Function "F" mapped onto the local erosion rate. Once the *new* geometry is generated it is exported to BOXERmesh and a *new* simulation mesh created and exported in turn to Fluent™. Figure 16 illustrates this simulation mesh (surface shown) for the eroded stator blade; particularly evident is the erosion pattern associated with the particulate vortical structure formed in the pressure side/ endwall corner. It is important to emphasize that this is a body-fitted mesh generated for a degraded geometry – *not* a morphed mesh – and that classical BREP NURBS CAD would have difficulty in representing this degraded geometry.

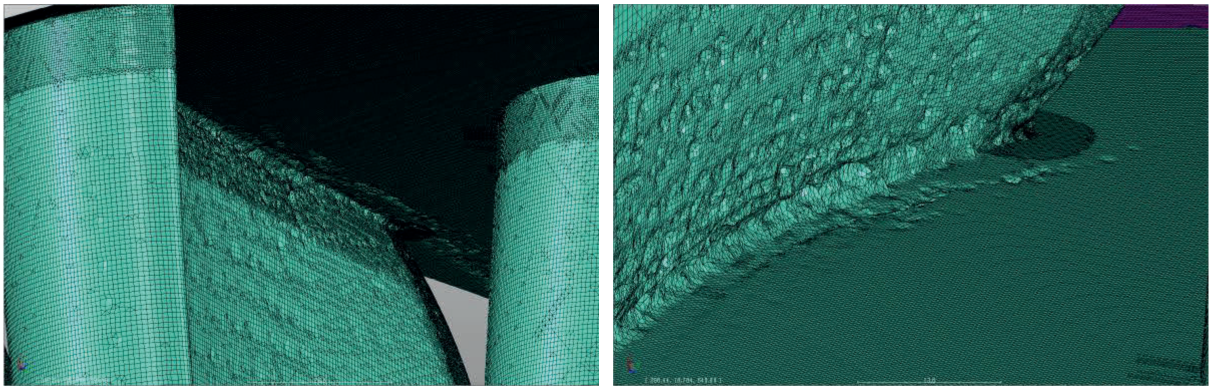

Fig.16: Simulation mesh (surface shown) for the eroded stator blade; particularly evident is the erosion pattern associated with the particulate vortical structure formed in the pressure side/endwall corner

A *new* flow simulation is run using Fluent™ and the result is illustrated in the next Figure which shows total pressures plotted in a constant streamwise cut near the stator vane TE plane for both the pristine and eroded geometries. The differences are relatively small but nevertheless the extra loss generated by the pressure side/endwall erosion is detectable.

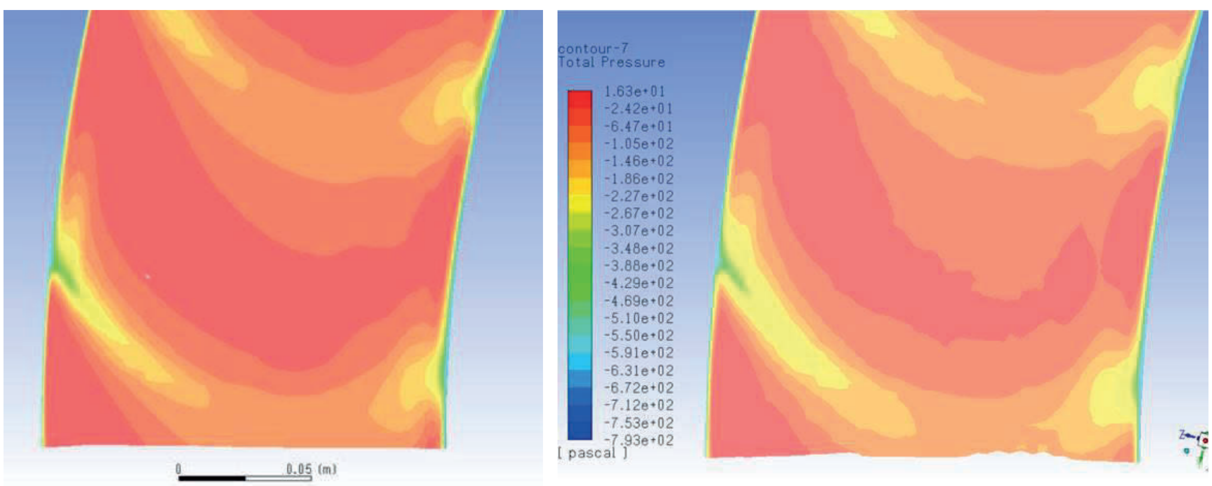

Fig.17: Predicted total pressures near the vane TE for the pristine (left) and eroded (right) geometries

To make this more quantitative, the Table shows the predicted loss coefficients for both pristine and eroded geometries together with an estimate of the impact of this on the Exhaust Gas Temperature (a key monitor of engine performance degradation) using the very simple analysis described in Dawes *et al* [2019] and for the nominal case of a by-pass turbofan engine at cruise (31,000ft, M=0.85, engine entry T_{02}=256K and cycle ratio T_{04}/T_{02}=6).

Stator Vane	Pristine	Eroded
Loss Coefficient, ξ	0.093	0.106
$\Delta\xi$	-	0.013
$\Delta\eta_T$	-	-0.013
ΔEGT	-	+22.1 °C
ΔEGT/1000FC	-	+5.1 °C

Table: Predicted loss coefficients and associated estimated changes in EGT for a nominal by-pass turbofan at cruise for the pristine & eroded geometries

As can be seen from this Table, the loss coefficient is significantly impacted during the simulated degradation event by the geometry changes. When imagined to be a blade in a by-pass turbofan at cruise, this increase in loss coefficient is estimated to cause a significant increase in EGT of 5.1 ^{o}C per 1000FC (1FC=2FH) – the magnitude compares well with Ackert [2011] who quotes 4-5°C per 1000FC.

This example has successfully demonstrated how physics-based simulation might be applied to estimate through-life performance changes associated with component geometry degradation.

B. In-service degradation: cooled turbine burn-through

The final example is the ambitious case of a simulated cooled turbine blade burn-through – respresentitive of the more extreme geometric perturbation illustrated earlier in Figure 7. As in the previous example, the “pristine” blade geometry, *Shape A*, comes from manufacturing CAD; the “damaged” geometry, *Shape B*, is created as a Point Cloud to mimic observed cases. Then, as before, we use Level-set Digital Geometry capability to morph “pristine” to “damaged” geometries thereby modelling the time history of the degradation, automatically exporting simulation meshes from BOXERmesh for CFD at selected stages during the morph-mesh-solve workflow. Currently we have used the classic Breen-Whitaker morph; next, the morph (and associated speed function “F”) we intend to take from a physics-based aero-thermal-mechanical-material simulation. Figure 18 shows the geometry during stages of the simulated burn-through of the cooled turbine blade.

Fig.18: Stages during the simulated burn-through of the cooled turbine blade: *geometry*

Figure 19 shows a detail of the blade leading edge geometry and associated mesh in the later stages of the burn-through simulation. The geometry is by now very complex but both the Digital Geometry model and associated mesh generation are perfectly happy handling this: every mesh exported from Boxer solves straightforwardly in Fluent™. We stress again the near impossibility of supporting this simulation with either BREP-NURBS geometry modelling or mesh morphing.

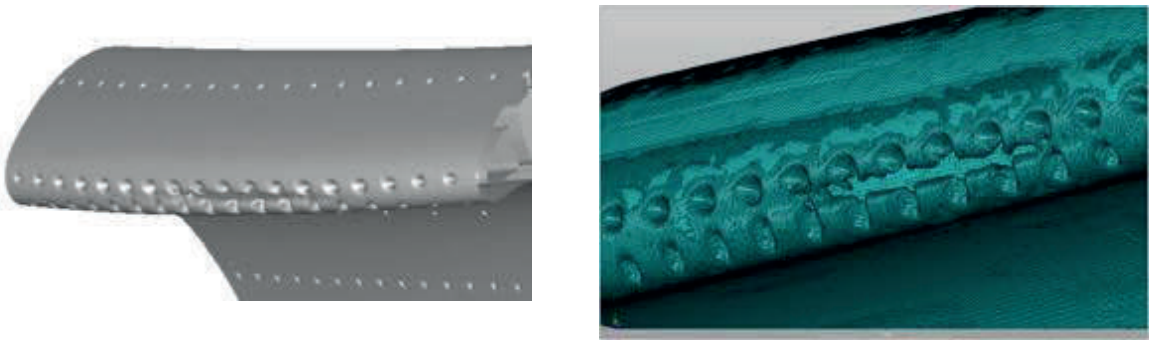

Fig.19: *Geometry* and associated *mesh* in the later stages of the burn-through simulation.

Finally, Figure 20 shows flow simulation results at stages during the simulated burn-through of the cooled turbine blade. The progressive – and increasingly serious – degradation of the performance is clear. In terms of stage matching and coolant load: as the throat opens, there is more primary mass flow; as the coolant holes "burn" there is more coolant flow also (fixed pressure ratio boundary conditions).

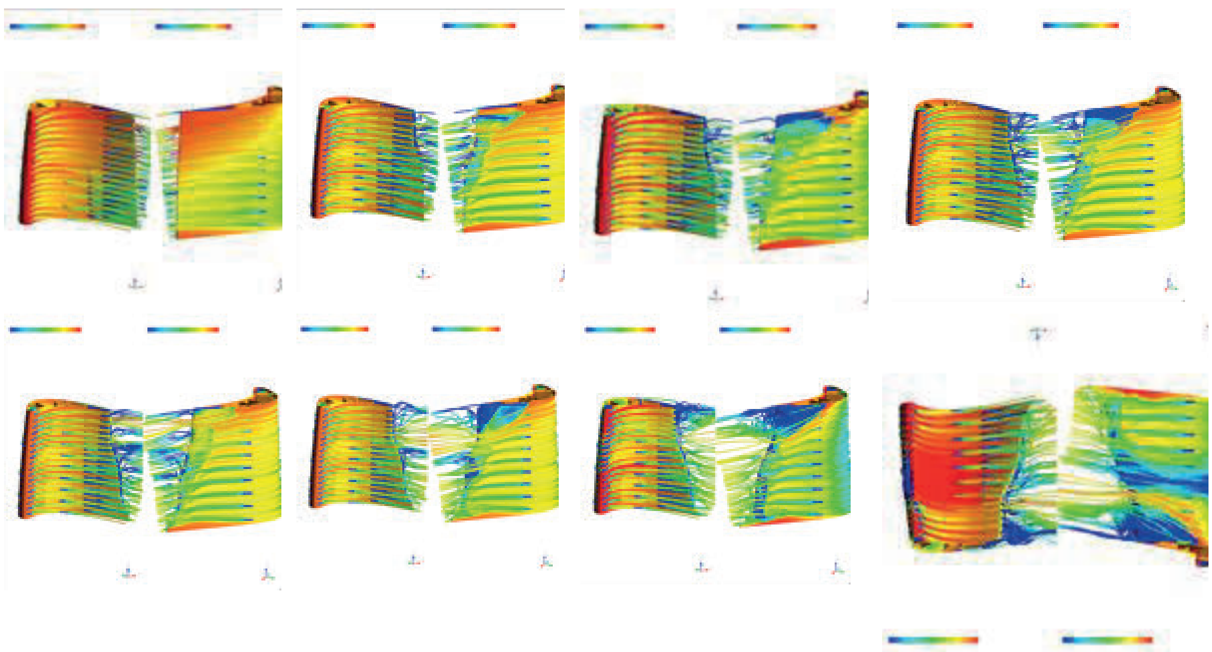

Fig.20: Stages during the simulated burn-through of the cooled turbine blade: *flow simulation*

VI. Concluding remarks

This paper has described a Gas Turbine Digital Twin and argued that its heart should be an integrated, physics-based simulation workflow (conjugate aero-thermal-mechanical plus manufacturing, erosion & corrosion, wear & degradation, cost & through-life economics).

Three key challenges were identified: scale of simulation; scaling the simulation; and responding to Data-driven feedback. Successfully confronting these challenges is needed to make the Digital Twin real.

The paper then argued that Geometry is the Digital Thread needed to make the Digital Twin real – and that a Digital Geometry model offers advanatges over the classical BREP-NURBS CAD approach. Some examples were given to illustrate the benefits of the proposed Digital approach.

As a final illustration, Figure 21 shows our perspective on geometry through the stages in the life of a product.

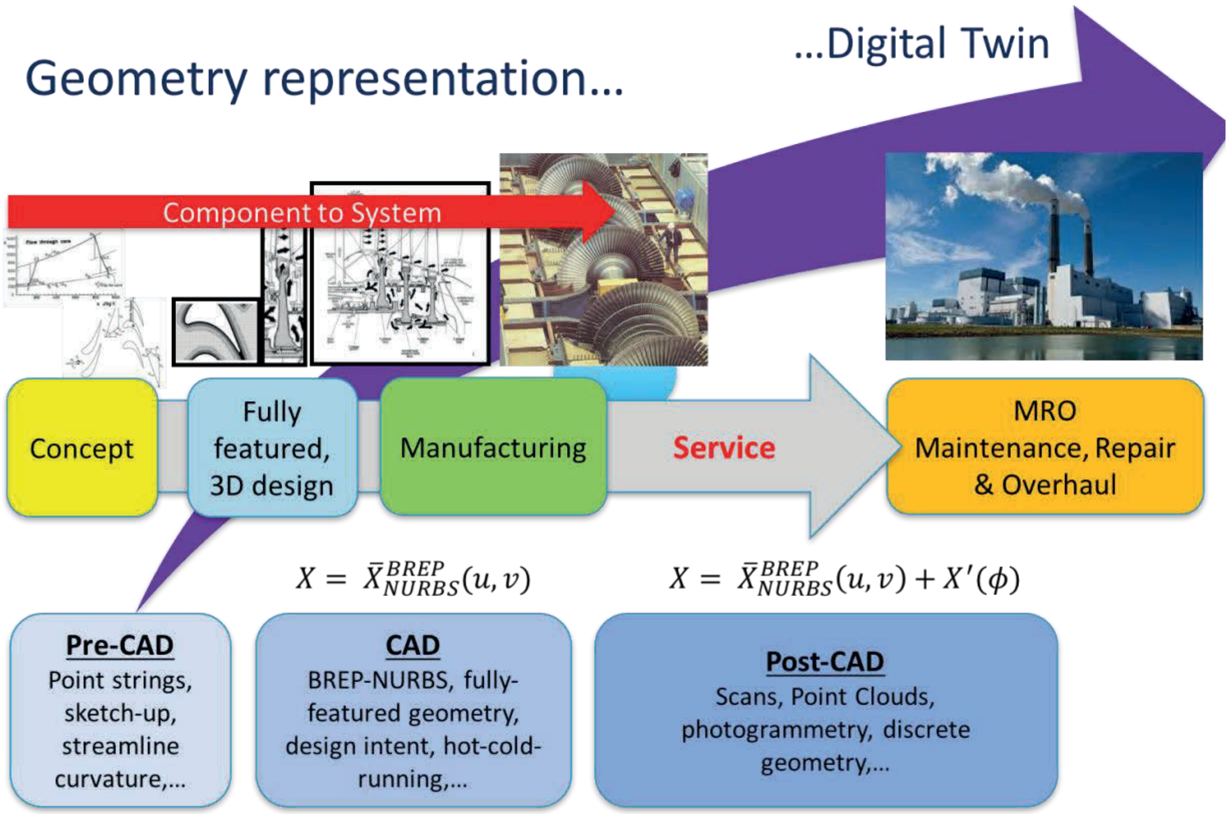

Fig.21: Geometry through the life of a product

Acknowledgements

We are very pleased to acknowledge partial financial support from Innovate UK via the GHandI, GEMinIDS and AuGMENT Consortia and also from our Development Partners. The authors are grateful to Cambridge Flow Solutions Ltd. for permission to publish this paper.

References

Ackert S “Engine Maintenance Concepts for Financiers” www.aircraftmonitor.com, Second Edition, Sept.2011

Adalsteinsson D & Sethian JA, “A level set approach to a unified model for etching, deposition & lithography II: three dimensional simulations” J.Comput.Phys, 122, pp348-366, 1995

Amdahl GM “Validity of the single processor approach to achieving large-scale computing capabilities” AFIPS Conference Proceedings (30): 483-485, 1967

Batalha, Eucledes da Conceicao Pereira “Aircraft Engines Maintenance Costs and Reliability” Masters Thesis, University of Lisbon, 2012

Bonilla CH “The effect of film cooling on nozzle guide vane ash deposition” MSc Thesis, Ohio State University (Mike Dunn), 2012

Breen DE & Whitacker RT “A Level-Set approach for the metamorphosis of solid models” IEEE Transactions in Visualisation & Computer Graphics, vol.7, no.2, April-June 2001

www.nondot.org/sabre/Mirrored/.../gpbb35.pdf Chapter 35: Bresenham is Fast and Fast is Good

Casari N, Pinelli M, Suman M, Montomoli F & di Mare L “EBFOG: deposition, erosion and detachment on high pressure turbine vanes” ASME Paper GT2017-64526, Charlotte NC, 2017

Dawes WN “Building Blocks Towards VR-Based Flow Sculpting” 43rd AIAA Aerospace Sciences Meeting & Exhibit, 10-13 January 2005, Reno, NV, AIAA-2005-1156

Dawes WN, Kellar WP, Harvey SA “Viscous Layer Meshes from Level Sets on Cartesian Meshes” 45th AIAA Aerospace Sciences Meeting & Exhibit, 8-11 January 2007, Reno, NV, AIAA-2007-0555

Dawes WN, Kellar WP, Harvey SA ”A practical demonstration of scalable parallel mesh generation” 47th AIAA Aerospace Sciences Meeting & Exhibit, 9-12 January 2009, Orlando, FL, AIAA-2009-0981

Demargne AAJ, Evans ROE, Tiller PJ & Dawes WN “ Practical and Reliable Mesh Generation for Complex, Real-world Geometries” AIAA-2014_0199

Dawes WN, Kellar WP, Harvey SA “Towards topology-free optimisation: an application to turbine internal cooling geometries” 46th AIAA Aerospace Sciences Meeting & Exhibit, 7-10 January 2008, Reno, NV, AIAA-2008-0925

Dawes WN, Kellar WP, Verdicchio J, Racz T, Meah N, Kudryavtsev A, Evans RO, Hunt M, Tiller M, Vasco C, Lu Yi & Kai Liu “Contributions Towards the NASA 2030 Vision for Simulation ” SciTech AIAA-2018-xxx, Kissimmee, FL, 2018

Dawes WN, Hunt M, Meah N, Kudryavstev A & Evans RO “Physics-based simulation in support of a through-life gas turbine service business model” ASME2019-90882 – submitted for publication

Dunn MG, Baran AJ & Miatech J “Operation of gas turbine engines in volcanic ash clouds” ASME Paper 94-GT-170, The Hague, June 1994

https://www.forbes.com/.../what-is-digital-twin-technology-and-why-is-it-so-important, March 2017

Finnie I, Stevick GR & Ridgeley JR “The influence of impingement angle on the erosion of ductile metals by angular abrasive particles” Wear, 152, pp91-98, 1992

Forsyth P, Gillespie DRH & McGilvray M “Development and application of a coupled particle deposition dynamic mesh morphing approach for the numerical simulation of gas turbine flows” ASME Paper GT2017-63295, Charlotte NC, June 2017

Gammon M, Bucklow H & Fairey R “A review of common geometry issues affecting mesh generation” AIAA SciTech 2018-1402, Kissimmee FL, Jan.2018

Graham LJW, Lester D & Wu J “Slurry erosion in complex flows: experiment & CFD” Seventh Internatinal Conference on CFD in the Minerals & Process Industries, CRIRO, Melbourne, 9-11 Dec., 2009

Haimes R & Dannenhoffer III JF “EGADSlite: a lightweight geometry kernel for HPC” SciTech AIAA-2018-1401, Kissimmee, FL, 2018

Hamed AA, Tabakoff W, Rivir RB, Das K & Arora P “Turbine blde surface deterioration by erosion” ASME J.of Turbo., 127, pp 445, July 2005

Lee WY, Dawes WN, Coull JD & Goenaga F, “The impact of manufacturing variability on high pressure turbine profile loss” AIAA SciTech, Kissimmee, 8-12 January 2018

https://blogs.sas.com/content/iml/2012/02/15/what-is-mahalanobis-distance.html

Meher-Homji CB, Chaker MA & Motiwala HM “Gas turbine performance deterioration” Proceedings of the 30th Turbomachinery Symposium, Texas A&M, 2002

Slotnick J *at al* “CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences” NASA/CR-2014-218178

Tabakoff W, Hamed A & Metwally M “Effect of particle size distribution on particle dynamics and blade erosion in axial flow turbines” ASME Paper 90-GT-114 , 1990

Wing RG & McGill IR “The protection of gas turbine blades” Platinum Metals Review, 25, (3), 94-105, 1981

-oOo-