Direct Coupling of a Two-Dimensional Fan Model in a Turbofan Engine Performance Simulation

by

Templalexis[1], Alexiou[2], Pachidis[3], Roumeliotis[4], Aretakis[2]

[1]Hellenic Air Force Academy

[2]National Technical University of Athens

[3]Cranfield University

[4]Hellenic Naval Academy

Contents

Contents

INTRODUCTION

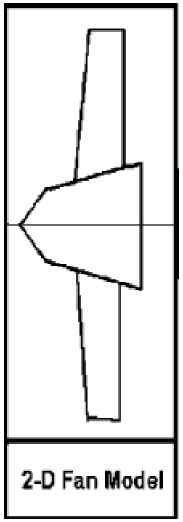
Component Zooming: execution of higher order analysis code and integration of its results back in the 0-D engine cycle allows for:

- more accurate physics & geometry based estimates of component performance
- complex phenomena & component design change studies

Different methods exist:

1. De-Coupled Approach
2. Semi-Coupled Approach
3. Fully-Coupled Approach

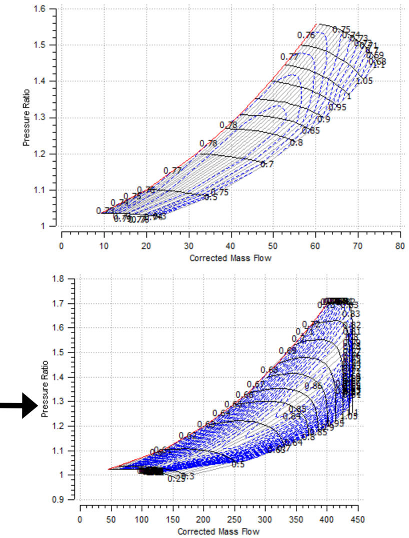
De-Coupled Zooming Approach



Run multi-point steady-state simulation to generate map data

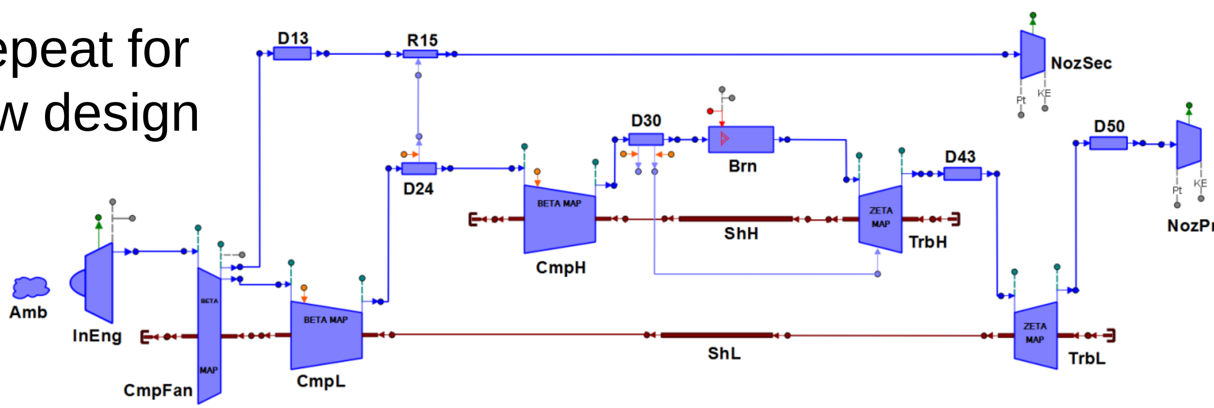
1.00000	20	20
1.24465	0.98896	0.58455
1.27283	0.98896	0.63243
1.30102	0.98896	0.67531
1.32920	0.98896	0.71529
1.35739	0.98896	0.75045
1.38537	0.98872	0.78269
1.41275	0.98783	0.80937
1.43973	0.98660	0.83112
1.46556	0.98432	0.84649
1.48894	0.97987	0.85445
1.50898	0.97263	0.85601
1.52595	0.96317	0.85445
1.53806	0.95009	0.84520
1.54629	0.93451	0.83070
1.55080	0.91670	0.81316
1.55151	0.89670	0.79298
1.54774	0.87399	0.76732
1.54072	0.84963	0.74247
1.53098	0.82403	0.71596
1.52145	0.79954	0.68860

Construct appropriate map format

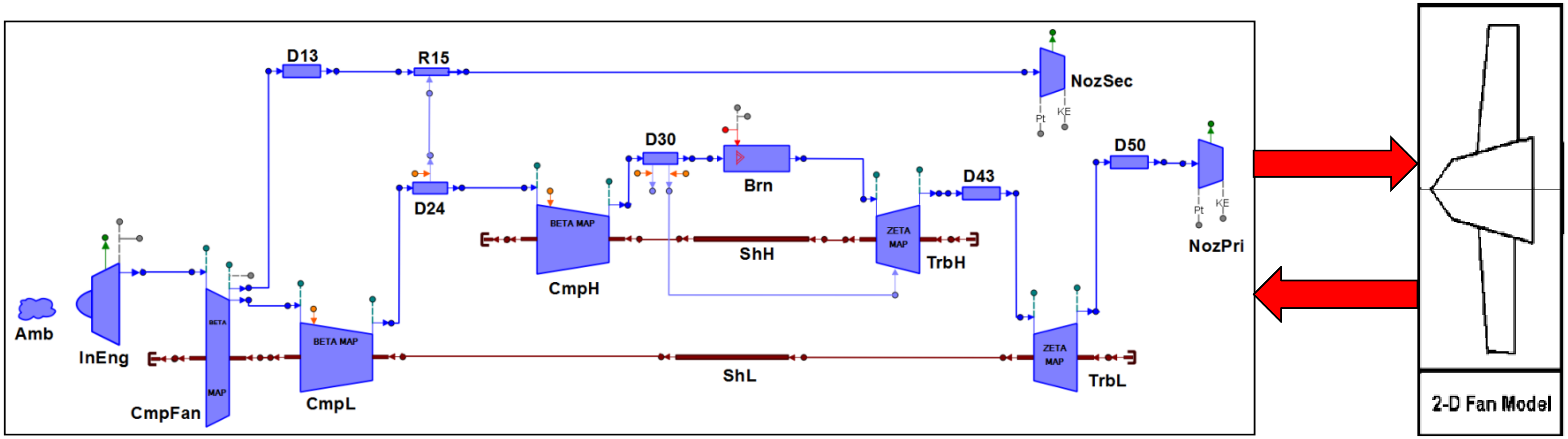


Select maps at component level & perform 0-D engine calculation

Repeat for new design

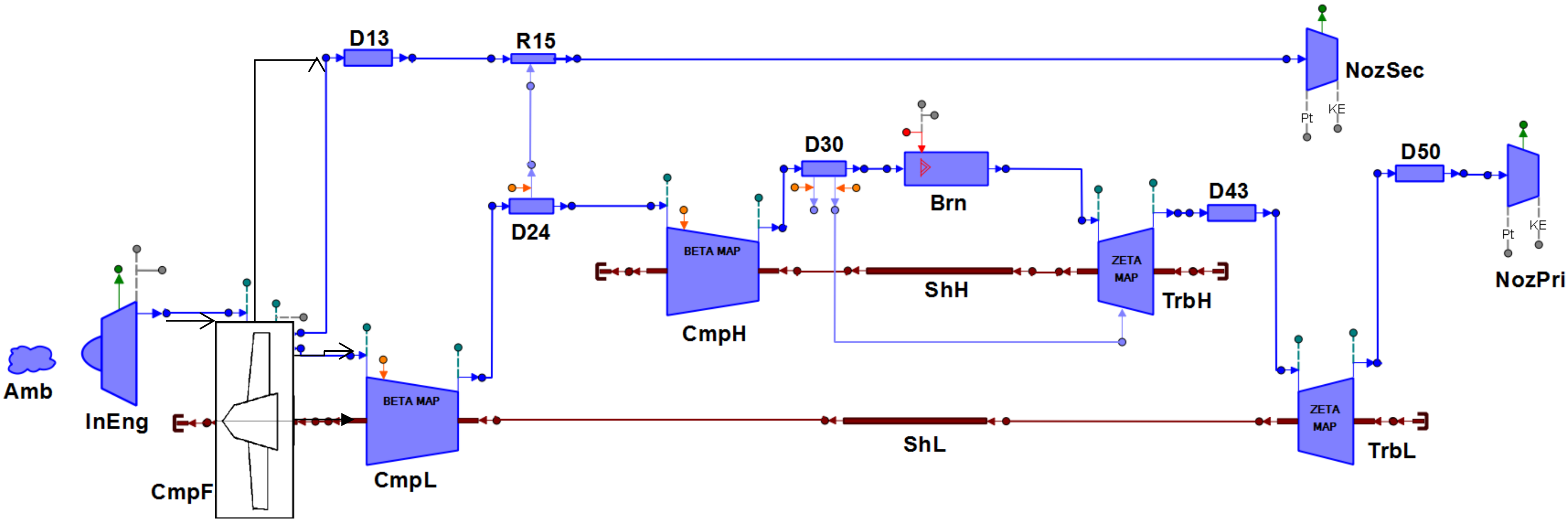


Semi-Coupled Zooming Approach



Run 0-D engine model → Pass 0-D component information to 2-D component → Run 2-D component model → Correct 0-D component performance accordingly through scalars

Fully-Coupled Zooming Approach



Remove 0-D component from engine model

Add 2-D component in its place

Connect 2-D component with engine model

Motivation & Objectives

- Have in hand a tool that can efficiently be used to address:
  - gas turbine installation effects (e.g. distorted inlet flow)
  - compressor/fan design optimization
  - engine health monitoring (through simulation of fouling/erosion/tip clearance)
- Build a computational tool combining an in-house Streamline Curvature (SLC) through flow solver with a commercial 0-D performance simulation tool.
- Follow the fully-coupled approach between the two solvers without affecting conventional model construction and simulation case definition.

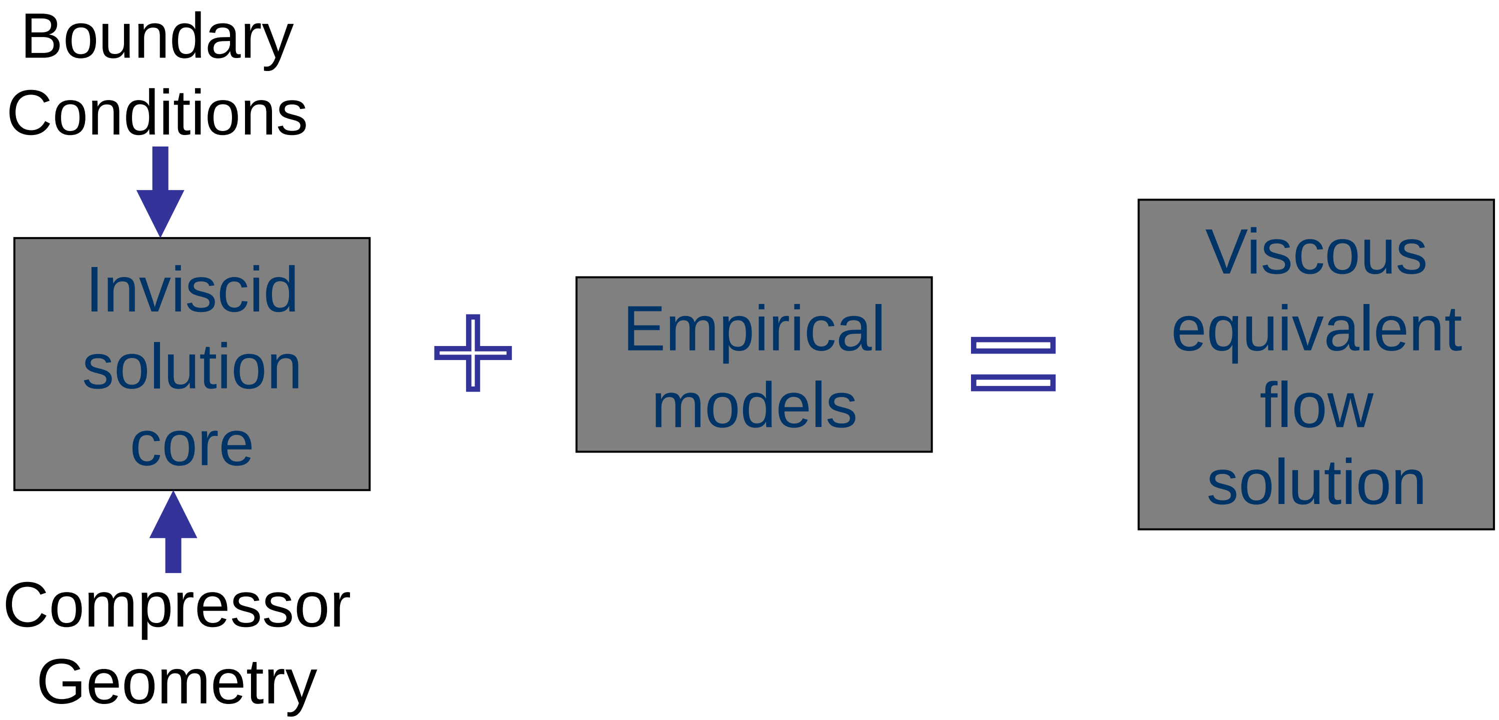
Contents

- ❑ INTRODUCTION
  - o Zooming Approaches
  - o Motivation & Objectives
- ❑ **HIGH FIDELITY FAN MODELLING**
  - o **Empirical Models**
  - o **Comparison with Experimental Data**
- ❑ 0-D ENGINE PERFORMANCE MODELLING
- ❑ MIXED FIDELITY ENGINE PERFORMANCE MODEL
  - o PROOSIS 2-D Fan Component
  - o Test Case
- ❑ SUMMARY & CONCLUSIONS

Through Flow Solver

SOCRATES: **S**ynthesis **O**f **C**orrelations for the **R**apid **A**ssessment of **T**urbomachine **E**ngine **S**ystems

Produces two dimensional flow solution based on Streamline Curvature (SLC) method.



SOCRATES details

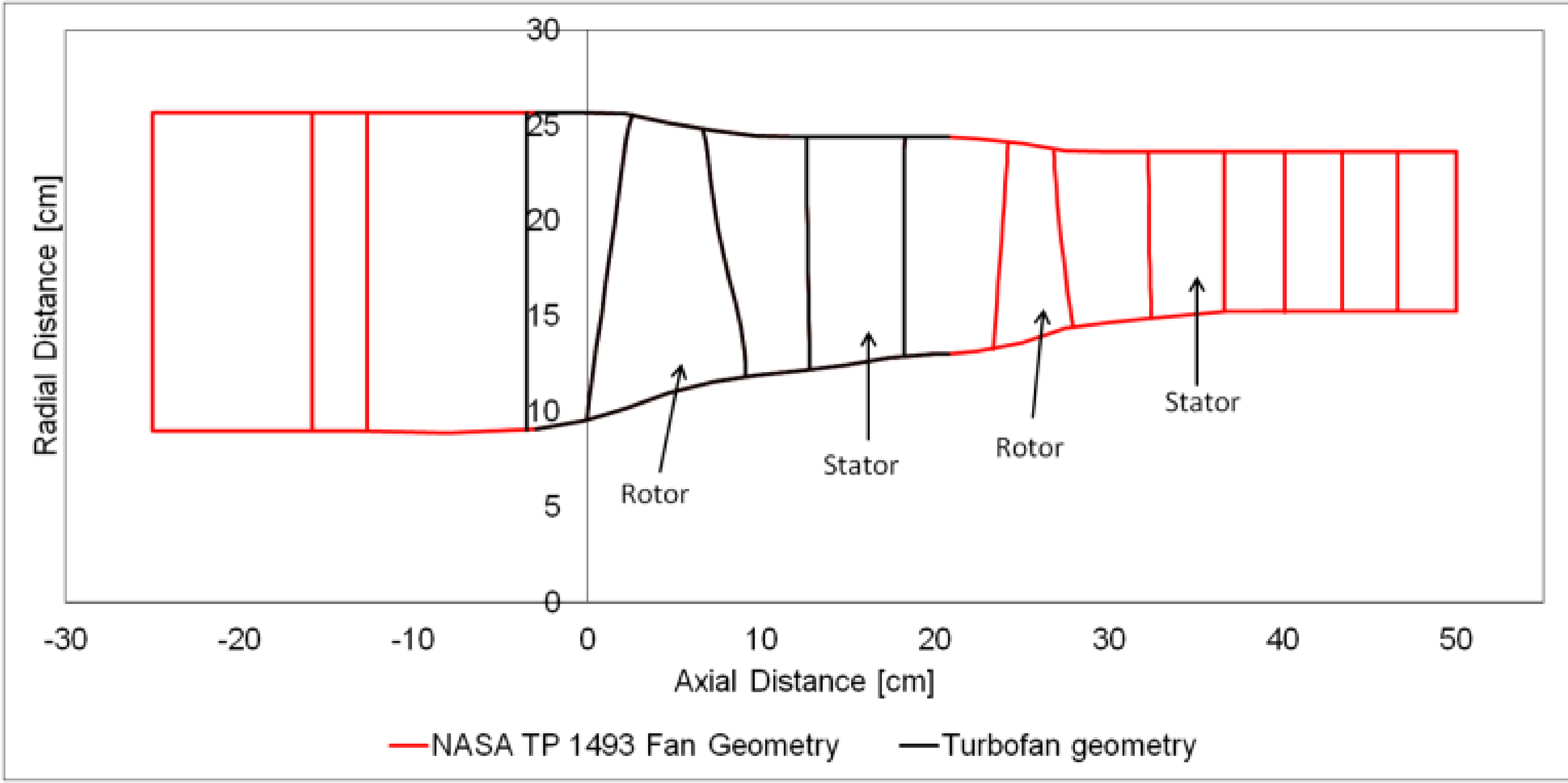
Inviscid Solution

- ✓ Full radial equilibrium equation considered along quasi-normal curves.
- ✓ Pseudo force terms included.
- ✓ Incorporation of dynamic convergence schemes with variable tolerance.

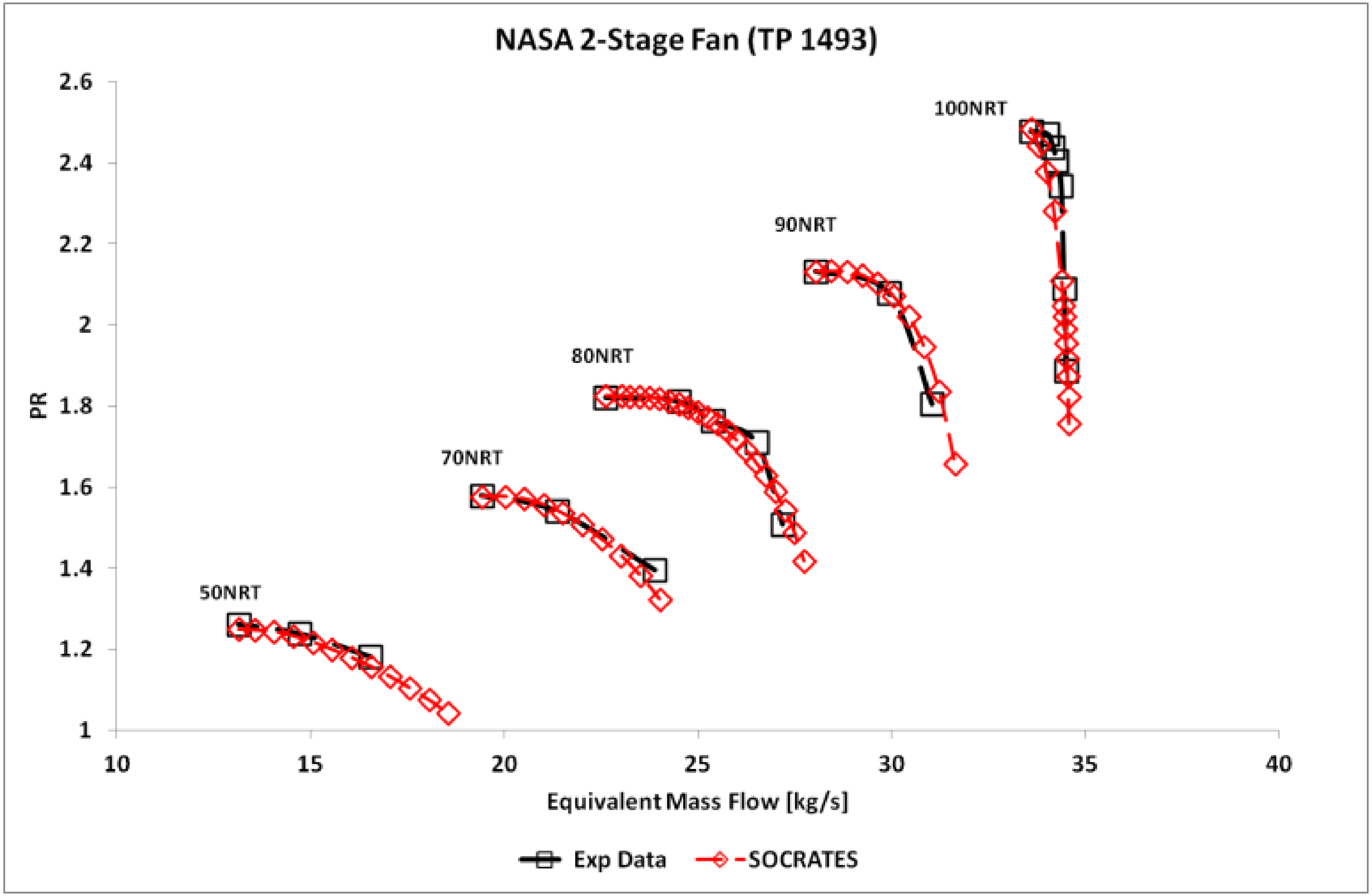
Empirical Models

- ✓ <u>Profile loss:</u> Bucket shaped curves (incidence vs profile loss) positioned based on flow Mach number level (Aungier)
- ✓ <u>Shock loss:</u> Simplified loss model based on frozen shock pattern (Swan).
- ✓ <u>Deviation angle:</u> Separate individual components considered due to off – design operation superimposed on the minimum loss value (Aungier)
- ✓ <u>Boundary layer:</u> Simplified end wall boundary layer model. Boundary layer shape parameters are defined from an integral approach based on meridional velocity along the boundary layer edge (Jansen & Moffatt)

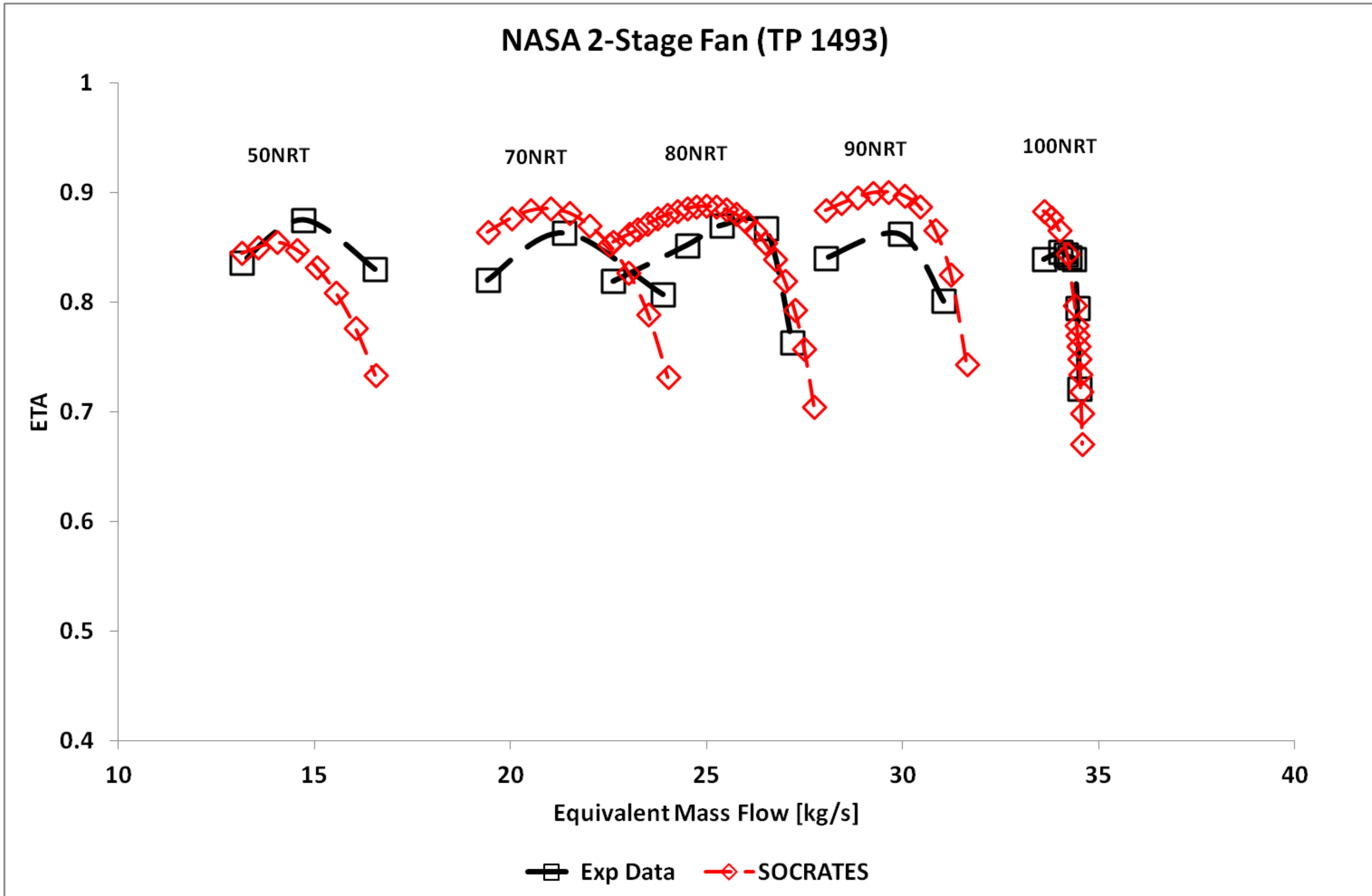
Validation – Geometry



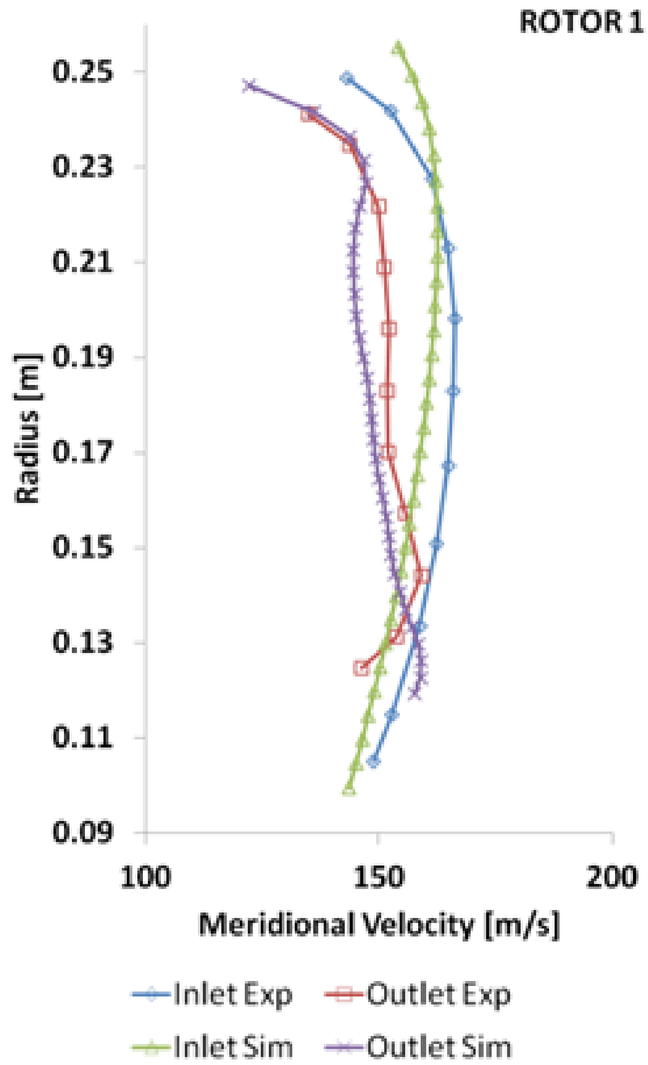
Validation – Performance Maps

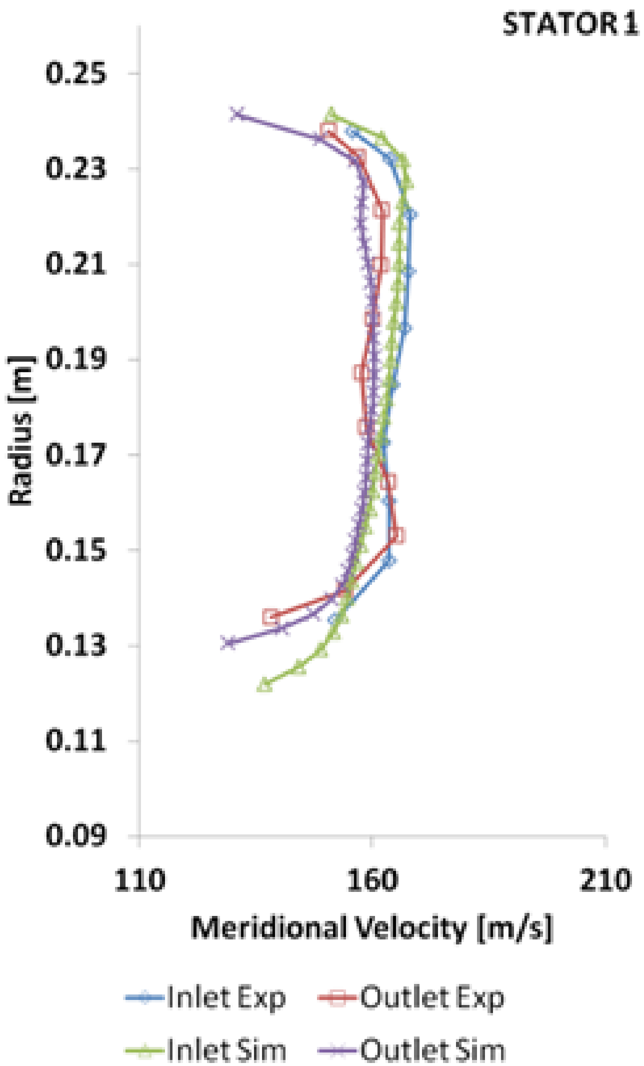


Validation – Performance Maps



Validation - Velocity Profiles





Socrates – Input/Output

Input

- number of flow stream lines.
- number of points when defining a radial profile of a property at component inlet.
- number of stages.
- number of stator and rotor blades of each stage.
- radial location (as a percentage) of points in a profile.
- the stator/rotor stagger angle radial distributions.
- the inlet air mass flow rate. / outlet static pressure
- the fan rotational speed.
- the inlet flow temperature, pressure and angle radial distributions.

Output

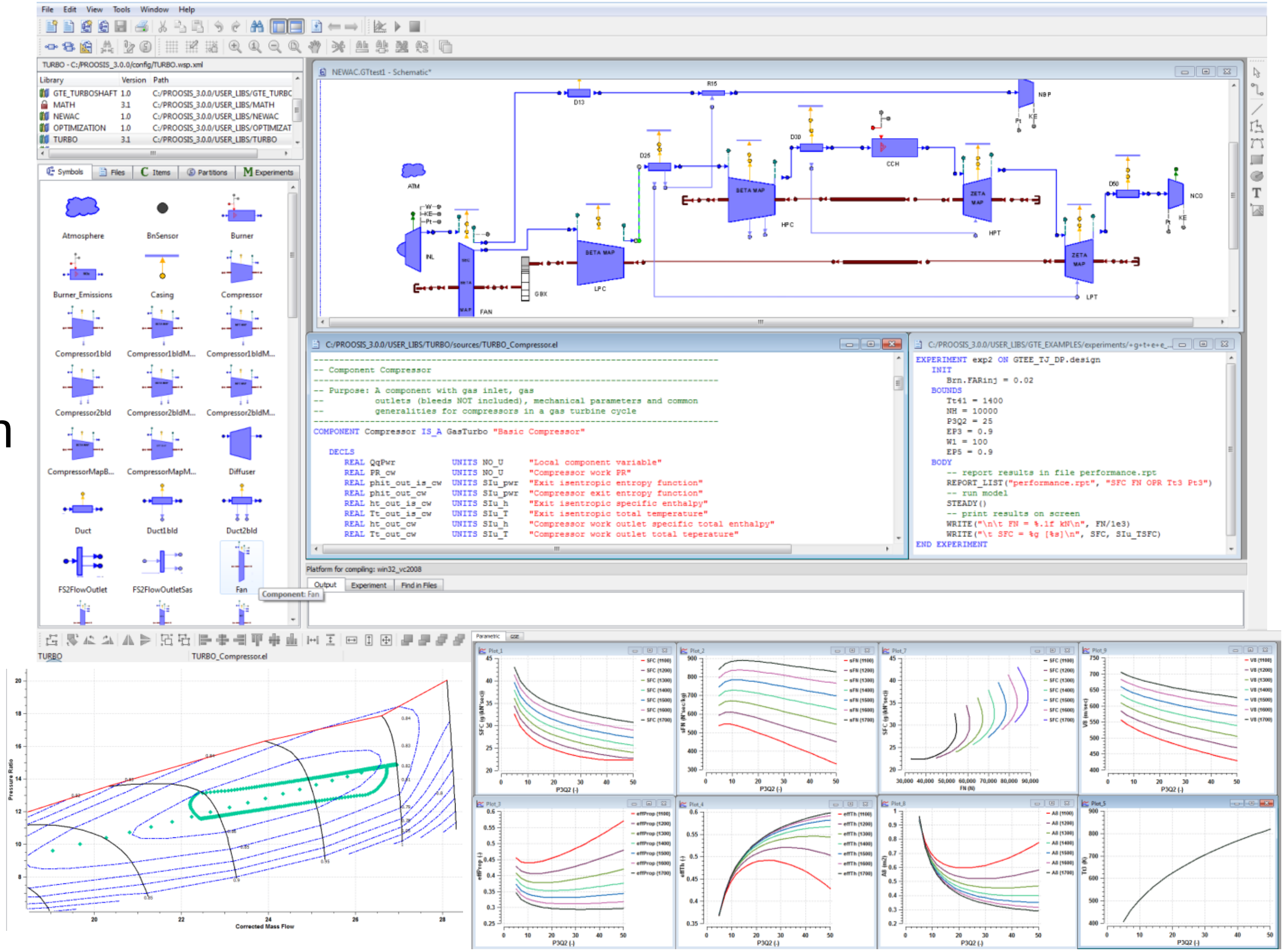
- Radial distribution of pressure.
- Radial distribution of temperature. number of stages.
- Radial distribution of meridional velocity component.
- Radial distribution of absolute velocity.
- Radial distribution of axial Mach number.
- Radial distribution of absolute Mach number
- Radial distribution of density.
- Radial distribution of flow angle
- Streamtube mass flow rate
- Streamtube cross sectional area.
- A convergence flag is also returned to warn the user in case of non-convergence

Contents

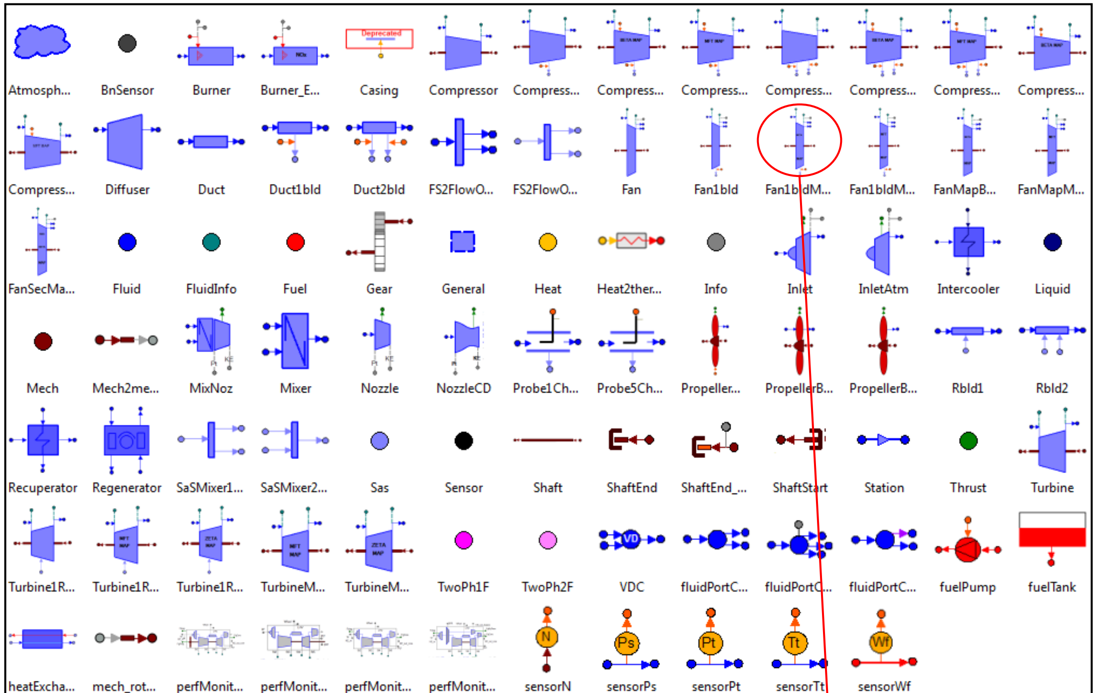
Simulation Platform

PROOSIS (PRopulsion Object-Oriented SImulation Software)

- Object-Oriented
- Steady State
- Transient
- Mixed-Fidelity
- Multi-Disciplinary
- Distributed
- Multi-point Design
- Off-Design
- Test Analysis
- Diagnostics
- Parametric
- Sensitivity
- Optimisation
- Deck Generation

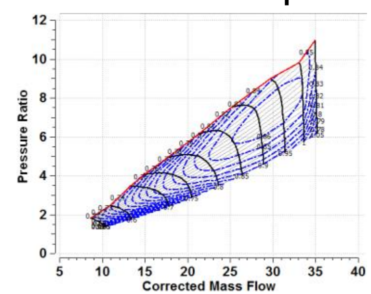


Model Construction

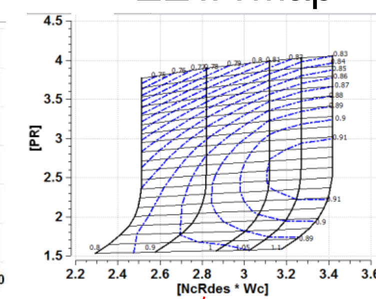


TURBO library of Gas Turbine Engine Components

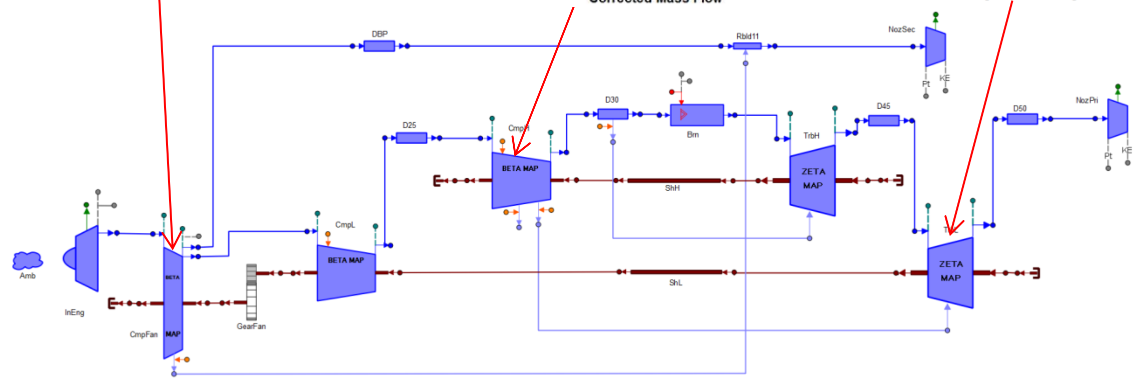
Compressor BETA map



Turbine ZETA map

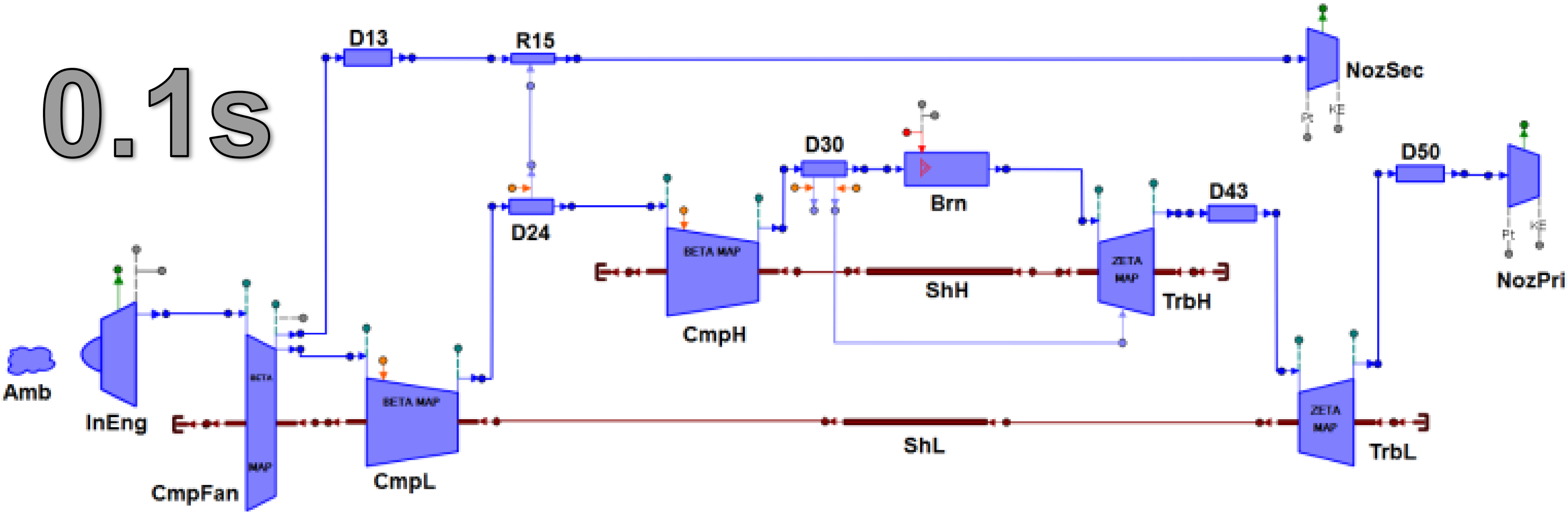


Engine configuration constructed graphically



Turbofan Engine Model

0.1s



<u>Mathematical Model</u>

820 equations + 410 input data

1 boundary variable (fuel flow rate)

8 algebraic variables (inlet flow rate, BPR, BETA/ZETA)

2 dynamic variables (shaft rotational speeds)

10×10 Jacobian matrix

Contents

SOCRATES in PROOSIS

SOCRATES from executable stand-alone application
→ subroutine with specific interface as static library

SOCRATES defined in PROOSIS as external FORTRAN function

```
"FORTRAN" FUNCTION NO_TYPE SOCRATES_SLL          (
        IN INTEGER NoOfBoundaryPointsIn,
        IN INTEGER NoOfTimeStepsIn,
        IN REAL SplitterRadiusIn,
        ...
        OUT REAL StrlineRadialPositionOut[],
        OUT REAL CompressorAbsTotPressureRatioOut,
        OUT REAL CompressorIsentrEffOut,
        ...
        ) IN "SOCRATES_SLL.lib"
```

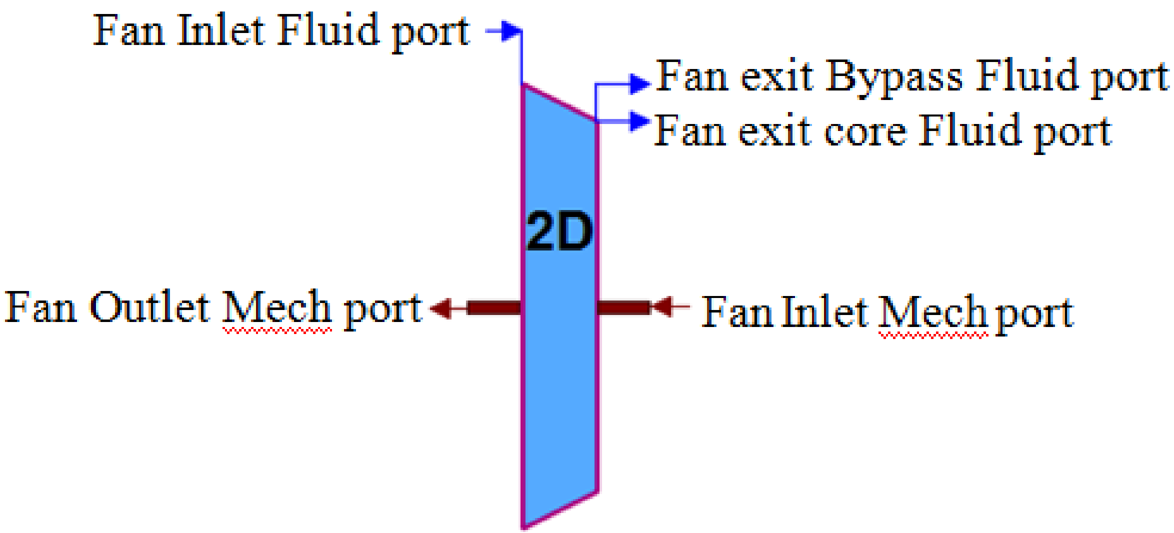
PROOSIS 2-D Component

Simple 2-D component definition based on PROOSIS abstraction and inheritance capabilities

```
Component Fan2D IS_A Fan

          SOCRATES_SLL(…)
          MassAverage (…)

END COMPONENT
```

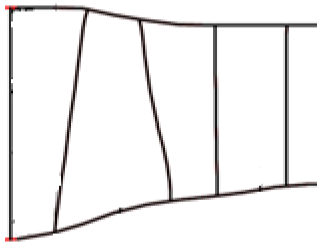


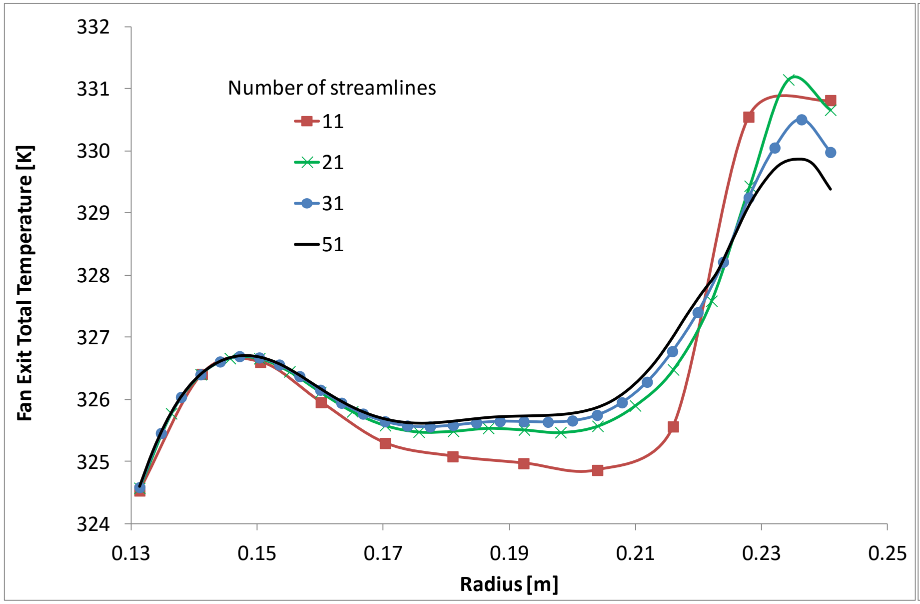
Use component to:

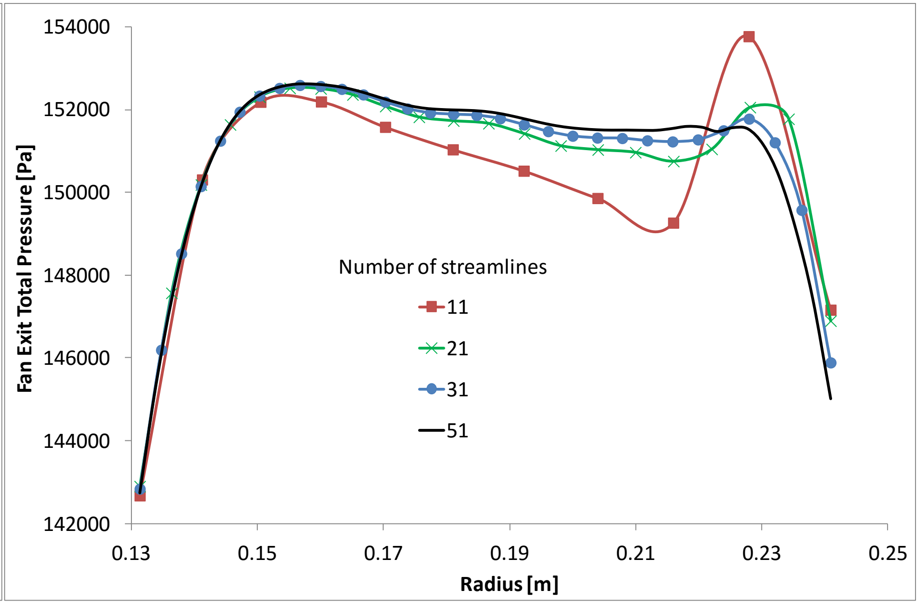
- Generate map
- Parametric studies
- Optimize geometry

Parametric Study – Effect of Streamline Number

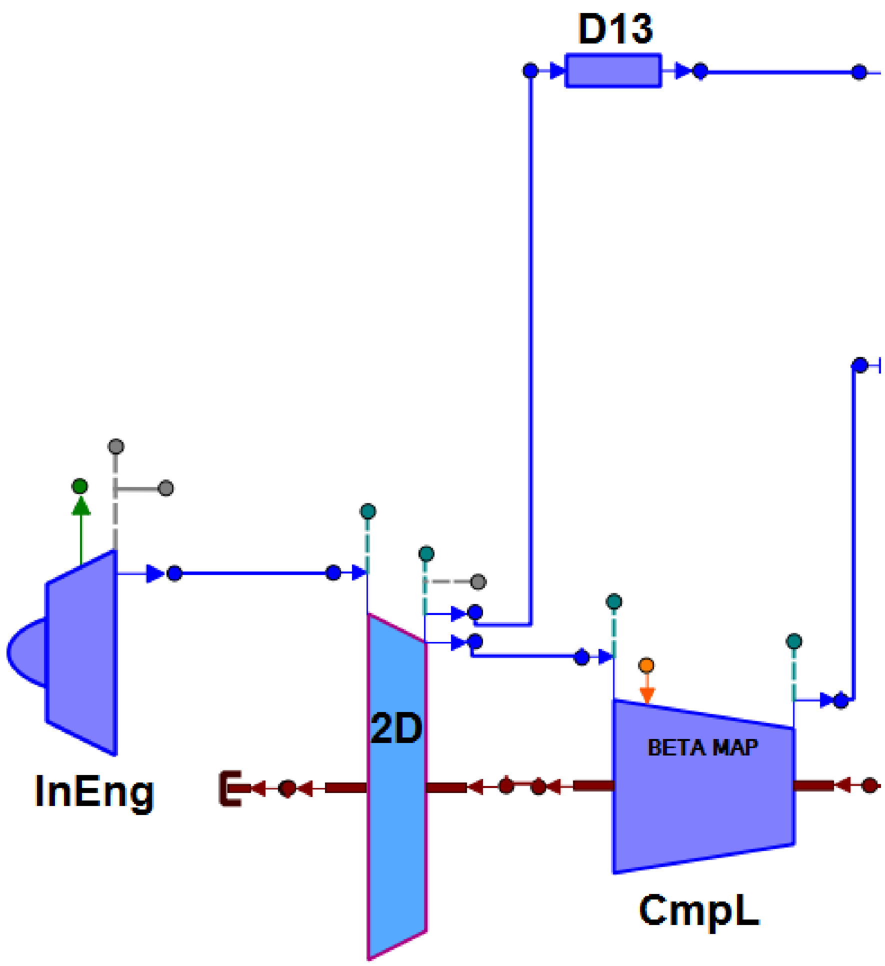
NASA TP1493 1st stage
SLS conditions
Win = 30 kg/s
N =14505 rpm







Mixed Fidelity Turbofan Model Construction



Mathematical Model
1 boundary variable
7 algebraic variables
2 dynamic variables
9×9 Jacobian matrix

~15min

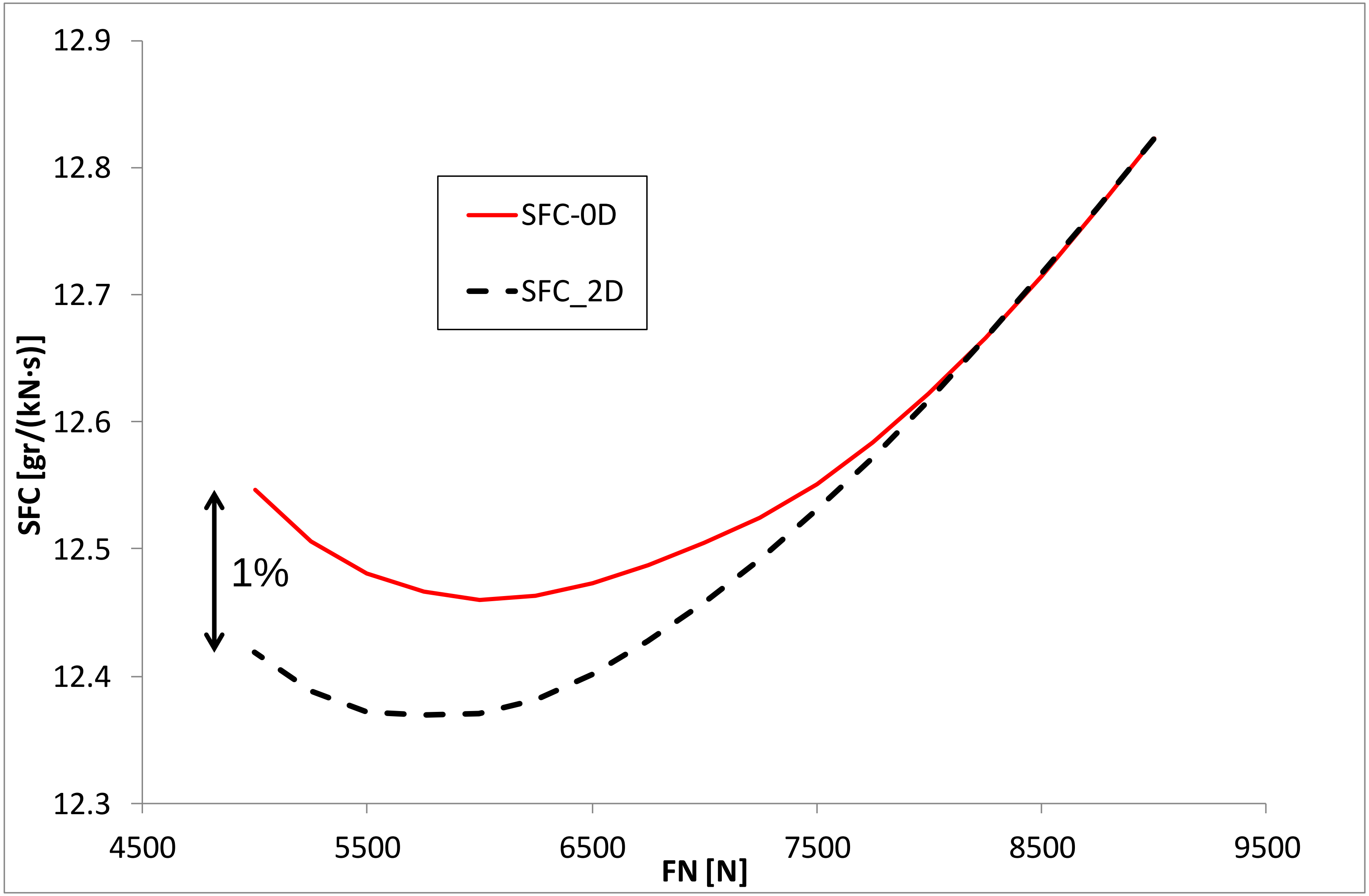
Test Case Design Point Definition (SLS)

Component	Parameter	Value
InEng	W [kg/s]	30
CmpFan	N [rpm]	14500
	BPR	3.3
	Inner fan eff [-]	0.914
	Inner fan PR [-]	1.492
	Outer fan eff [-]	0.900
	Outer fan PR [-]	1.500
CmpL	PR [-]	1.3
	eff [-]	0.87
CmpH	PR [-]	6.5
	eff [-]	0.85
	N [rpm]	35000
Brn	eff [-]	0.99
	Pressure loss [%]	3.5
TrbH	eff [-]	0.86
TrbL	eff [-]	0.88
Perf	FN [N]	9000

Test case Design Point Selected Results

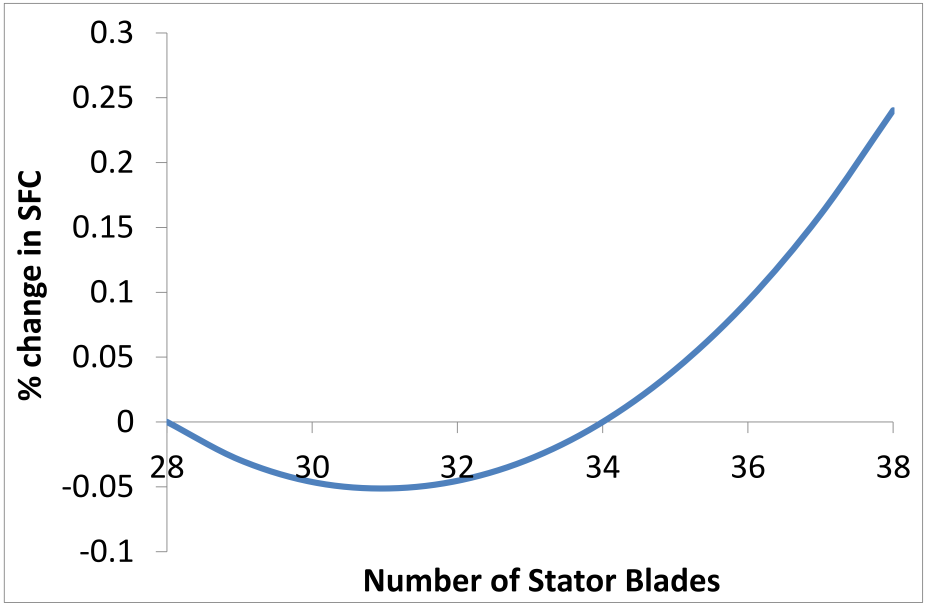
Parameter	Value
TrbH PR [-]	3.1
TrbL PR [-]	2.4
WF [kg/s]	0.115
TET [K]	1292
SFC [g/(kN·s)]	12.82

Operating Line for 0-D and 2-D Models

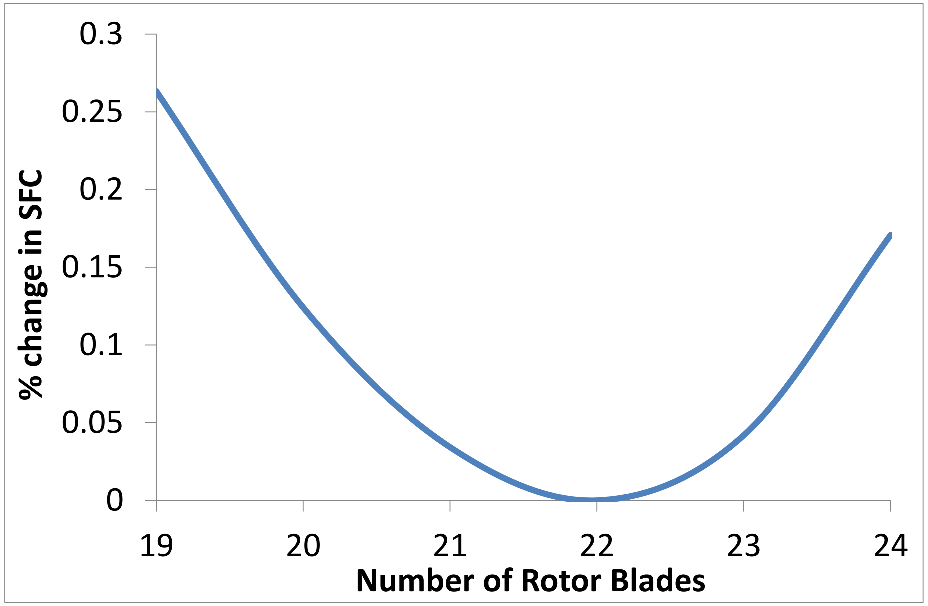


SFC Variation with Stator & Rotor Blade Number

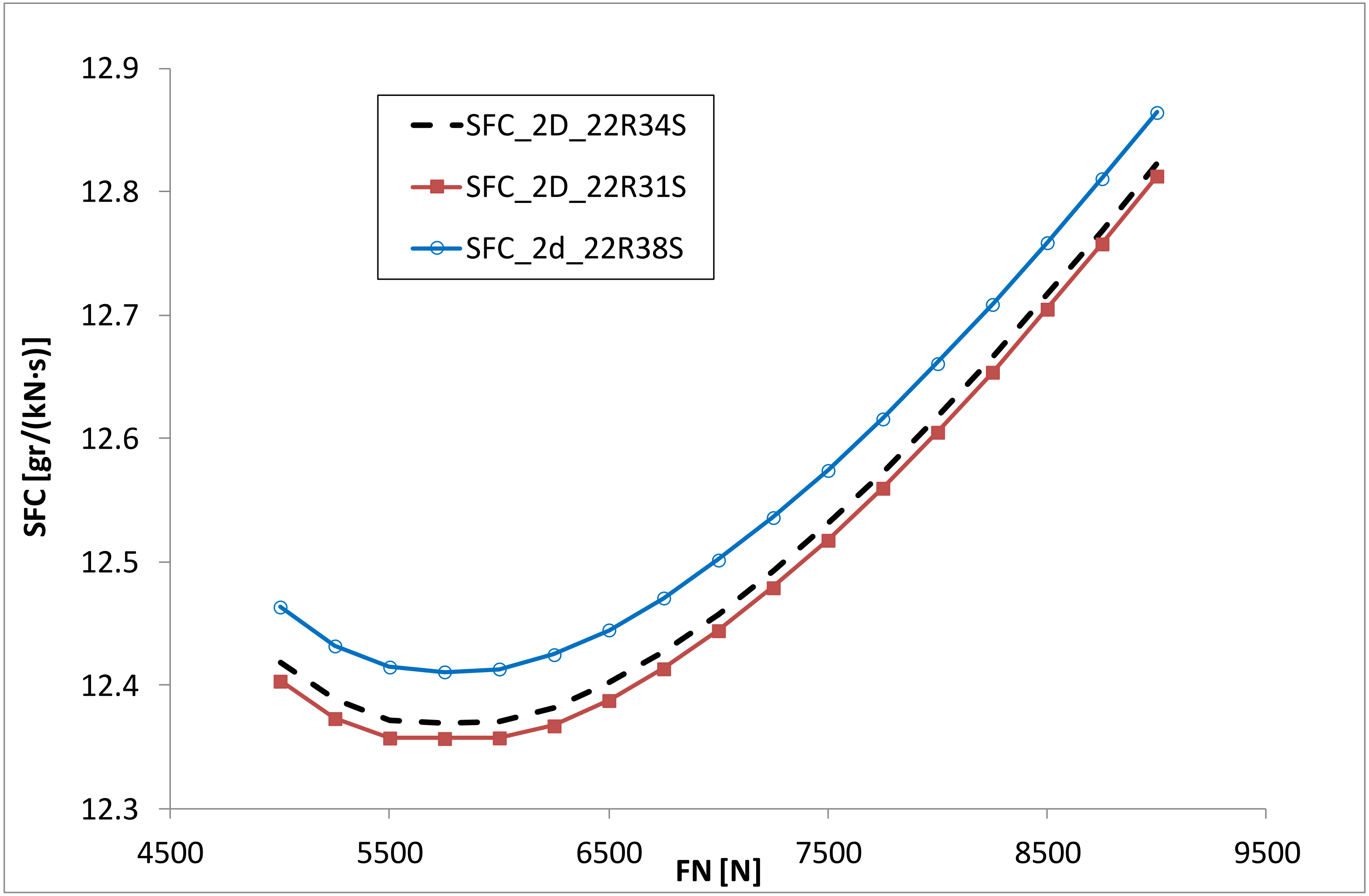
Stator = 34 blades



Rotor = 22 blades



Operating Line for Different Rotor-Stator Blade Numbers



Contents

Summary & Conclusions

- The integration of a stand-alone higher fidelity code in a 0-D engine performance simulation environment was presented through design point and off-design studies. The potential of the tool for advanced types of analysis is exemplified with a parametric study that calculates engine performance for different fan geometry settings.
- The proposed approach does not affect the model building procedure and the logic of the established mathematical formulation in existing performance simulations.
- The method presented does not depend on the fan geometry and type of engine or performance and can be extended to cover other engine components in an engine performance simulation.

감사합니다 – THANK YOU

Cranfield
UNIVERSITY