Koninklijk Nederlands Lucht- en Ruimtevaartcentrum
Royal Netherlands Aerospace Centre

GSP 12 User Manual

GSP Development Team
www.gspteam.com

Version 12.0.4.0 17-Mar-23

GSP 12 User Manual

by the GSP Development Team

NLR's Gas turbine Simulation Program (GSP) is an off-line component-based modeling environment for gas turbines. Both steady state and transient simulation of any kind of gas turbine configuration can be performed by establishing a specific arrangement of engine component models. GSP is a powerful tool for performance prediction and off-design analysis. GSP is especially suitable for sensitivity analysis of variables such as: ambient (flight) conditions, installation losses, certain engine malfunctioning (including control system malfunctioning), component deterioration and exhaust gas emissions.

GSP 12 User Manual

Publisher
NLR - Royal Netherlands Aerospace Centre
Anthony Fokkerweg 2
1006 BM Amsterdam
The Netherlands

Published
March 2023

Content editor
GSP Development Team

Information
www.gspteam.com

Table of Contents

1 Welcome

Gas turbine Simulation Program version 12 for MS-Windows

Authors	GSP Development Team members
Copyrights	Royal Netherlands Aerospace Centre - NLR
Home page	www.GSPteam.com
Help build nr	12.0.4.0
Help build date	Friday, March 17, 2023

2 Glossary

Abbreviation	Description	SI Dimension
A	Flow cross area	[m²]
Aratio	throat/exit area ratio	[-]
Athr_rel	A throat relative to A throat design (Athr/Athrdes)	[-]
BPR	Bypass ratio	[-]
bld	bleed flow	
c	core flow (fan, mixer)	
Cf	Correction factor for effective fan duct or core map mass flow	[-]
cl	cooling flow	
d	duct flow (fan, mixer)	
delta	pressure normalized to ISA = p[bar]/1.01325	
DP	Design Point (simulation/calculation mode)	
DHWin	change in H*Win	[kW]
DHWcl	shaft power produced by turbine cooling flow	[kW]
DHWkincl	shaft power needed to increase turbine cooling flow kinetic energy DHWcl due to rotation	[kW]
Droop	Proportional gain in rotor speed governor (P in PID control)	
dTs	Deviation from ISA standard temperature	[K]
Elco	carbon monoxide emission index	[g/(kg fuel)]
Elnox	nitrogen oxide emission index	[g/(kg fuel)]
Eluhc	unburned hydrocarbons emission index	[g/(kg fuel)]
Eta	efficiency	[-]
FAR	Fuel air ratio	[-]
Fbld	bleedfraction	[-]
FG	Gross thrust	[kN]
FN	Net thrust	[kN]
H	specific enthalpy	[kJ/kg]
Hbld	bleed flow specific enthalpy	[kJ/kg]
ISA	International Standard Atmosphere	
LHV	Lower Heating Value	[kJ/kg]
Mach, Macha	Flight mach number	[-]
Mexit	exit speed in Mach	[-]
N	rotor speed	[rpm]
Nc	corrected rotor speed	[rpm]
Ndm	demanded rotor speed	[rpm]
Nerror	rotor speed error signal in control system	[rpm]
Nsens	rotor speed signal sensed in control system	[rpm]
OD	Off-design (simulation/calculation mode)	
Pacc	accelleration power	[kW]
Pb	Pressure in combustor. "burner pressure"	[bar]
Pbld	Bleed flow pressure	[bar]
PLA	Power Lever Angle	[°]
PR	Pressure Ratio	[-]
PS, Ps	Static pressure	[bar]
PSexit	static exit pressure	[bar]
Pt, PT	total pressure	[bar]
PWshaft	Shaft power output	[kW]
RD	Ram drag	[kN]

RR	Ram Recovery factor	[RR]
SFC	Shaft Power Specific Fuel Consumption	[kg/(kW h)]
SN	smoke number	
Tbld	bleed flow temperature	[K]
Tcorr	corrected temperature	[K]
theta	temperature normalized to ISA = T[K]/288.15	
Time	time	[s]
TRQ	torque	[N m]
TSexit	static exit temperature	[K]
TSFC	Thrust Specific Fuel Consumption	[kg/(N h)]
TT	total temperature	[K]
Vc	Calibrated air speed	[m/s]
Vt	True air speed	[m/s]
W	air or gas mass flow rate	[kg/s]
Wbld	bleed mass flow rate	[kg/s]
Wc	corrected mass flow rate	[kg/s]
Wf	fuel flow	[kg/s]
WfPb	fuel flow devided by burner pressure	[(kg/s)/(bar)]
WfPblead	WfPb lead signal (differential term in PID control)	[(kg/s)/(bar)]
WfPbmax	Maximum acceleration schedule WfPb	[(kg/s)/(bar)]
WfPbmin	Minimum deceleration schedule WfPb	[(kg/s)/(bar)]
WfPbtrim	WfPb trim signal (integral term in PID control)	[(kg/s)/(bar)]
Zp	Pressure altitude	[m]

3 Document Conventions

Throughout the help guide the following conventions have been used:

- Normal font text is used to describe the help topics
  This text is an example of the used font.
- Links to other topics are underlined and colored
  This link shows an example. Note that the captions of the link does not always match the actual topic header for readability reasons.
- Fixed width font (courier) is used to indicate this is a name of a control of one of the controls used in either the GSP main application, the Project window, or any other input or data window.
  `This is an example of the font.`
- Some options are found as sub options of option groups in e.g. file menus. The vertical line separator (|) is used to separate two or more possible options in command syntax.
  `File|Reopen| .....`

4 Introduction

Gas turbine Simulation Program version 12 for Windows

Welcome to the Gas turbine Simulation Program GSP for Microsoft Windows. This GSP help manual (also available online through the GSP team website) includes information and guidelines for using the basic functionalities to build, adapt and run simulations of gas turbine models. For advanced users, detailed information on GSP modeling and simulation can be found in the GSP Technical Manual which is available to paid customers on request.

Royal Netherlands Aerospace Centre - NLR's Gas turbine Simulation Program (GSP) is a component-based gas turbine modeling environment. Both steady state and transient simulation of any gas turbine configuration can be performed by establishing a specific arrangement of engine component models. GSP is a powerful tool for performance prediction and both design point and off-design analysis and performance optimization. GSP is especially suitable for sensitivity analysis of variables such as: ambient (flight) conditions, installation losses, certain engine malfunctioning (including control system malfunctioning), component deterioration and exhaust gas emissions.

As of GSP version 11, a comprehensive case and configuration management functionality introduces features to improve usability, productivity, data integrity, and security. GSP contains session management, configuration and case management, process control, security features and improvements of the GUI. Note that due to the implementations of these functionalities restrictions apply to the setup of certain models. This implies that importing old version models (models saved in version 10 or lower) into the new project based structure may require some additional attention.

As of GSP version 12, a more powerful database storage solution is introduced. This SQL based storage solution offers comparison of various case results to facilitate much simpler analysis of case effects. Furthermore the output is better structured, grouped and colored per model component to find output parameters much more quickly. This version is also capable to adhere better to current standards as AS755 and ARP5571.

A great deal of GSP's power to rapidly analyse all sorts of engine performance aspects is due to the graphical user-interface, which parallels the consistent modular (i.e. "object oriented") structure of GSP. GSP's interface allows new users to perform easy "quick start" analysis tasks without the need to know all about the advanced features.

This ease-of-use is due to GSP being set up around three basic elements:

- Project window
- Component libraries as part of the main application window
- Result tables and results graphs

4.1 GSP versions and editions

GSP uses a specific version numbering scheme to identify the specific version of the program and any associated executable programs or modules.

GSP is available in two editions

- a Light Edition (LE) with limited functionality, free for non-commercial use.
- a fully functional professional Standard Edition (SE) for licensed users,

For the professional Standard Edition, a license is required. Contact NLR for obtaining or purchasing a license.

A variety of component libraries is available depending on the license type. The licensed version has a large number of additional components. Custom component libraries are available or can be developed for specific projects and customers and can be customer proprietary.

The Standard Edition can be extended with a Component Developers Package (CDP) license including source code that can be adapted and compiled to run with the GSP main program. This way, advanced users can adapt or extend component models to their own requirement. Be aware that this very advanced and requires thorough knowledge of Delphi (object Pascal), object-oriented programming and gas turbine theory and simulation.

See the Feature Matrix for details on functionalities and items included in the different products.

In GSP version 11.3.4.0, the GSP API (GSP Application Programming Interface) has been introduced. Usage of the API is restricted for non-registered users, it requires a dedicated GSP API license to use. A separate manual has been written for usage of the GSP API. As of GSP 12, the installer will install a limited GSP API (dll). This version allows 2 inputs and 3 output values, API registrants have the option to use unlimited input and output parameters.

4.1.1 Version numbering

GSP employs a version numbering scheme for both the main program executable and dynamically loaded libraries (.BPL files). The File version number string contains details on respectively the main version, 2 sub version (major and a minor) numbers, and the build number. For example version 11.0.2.8 means *main version* 11, *major version* 0, *minor version* 2 and *build number* 8. The version numbers are important information to be added to validation reports or when communicating with the GSP development team on bugs or other issues. The GSP development team will release stable release versions ending with a build number of value zero. Build numbers are used by the team to identify certain fixes, improvements or new features. However, intermediate releases, e.g. special client builds may differ from this numbering, i.e. a build number can be higher than zero.

The File version number shown in the About window is the main GSP executable (GSP.exe) build number. The main GSP executable includes all standard libraries.

Custom libraries are separate executables (BPL dynamically loaded) and therefore have separate build numbers, shown in the bottom status bas of the library window or docking panel. Normally these are issued/installed together with a new main GSP executable, but in special cases the custom library version numbers may be different, for example after installing an updated custom library BPL only (maintaining the same main GSP executable). Note that different build numbers may cause conflicts.

4.1.2 Light Edition

In the GSP 12 LE edition (free unregistered version), all standard gas path and control components are fully functional. However, saving to the project file of user specified data of a number of advanced components is inhibited. During a project session all data are still saved in the project tree configurations and cases (enabling inheritance of configurations and cases) but these data are not saved into the project file so have to be re-entered in the component data entry windows when re-opening the project file. Register your copy of the Light Edition to make it fully functional.

The LE edition is for non-profit use only and does not include access to GSP team support options.

GSP LE can be identified by the **LE/Light Edition** red mark in the About box and in the component data entry windows (lower left) for which saving of user specified data into the project file is inhibited.

The GSP 12 Light Edition version can be downloaded from the NLR GSP homepage at www.gspteam.com.

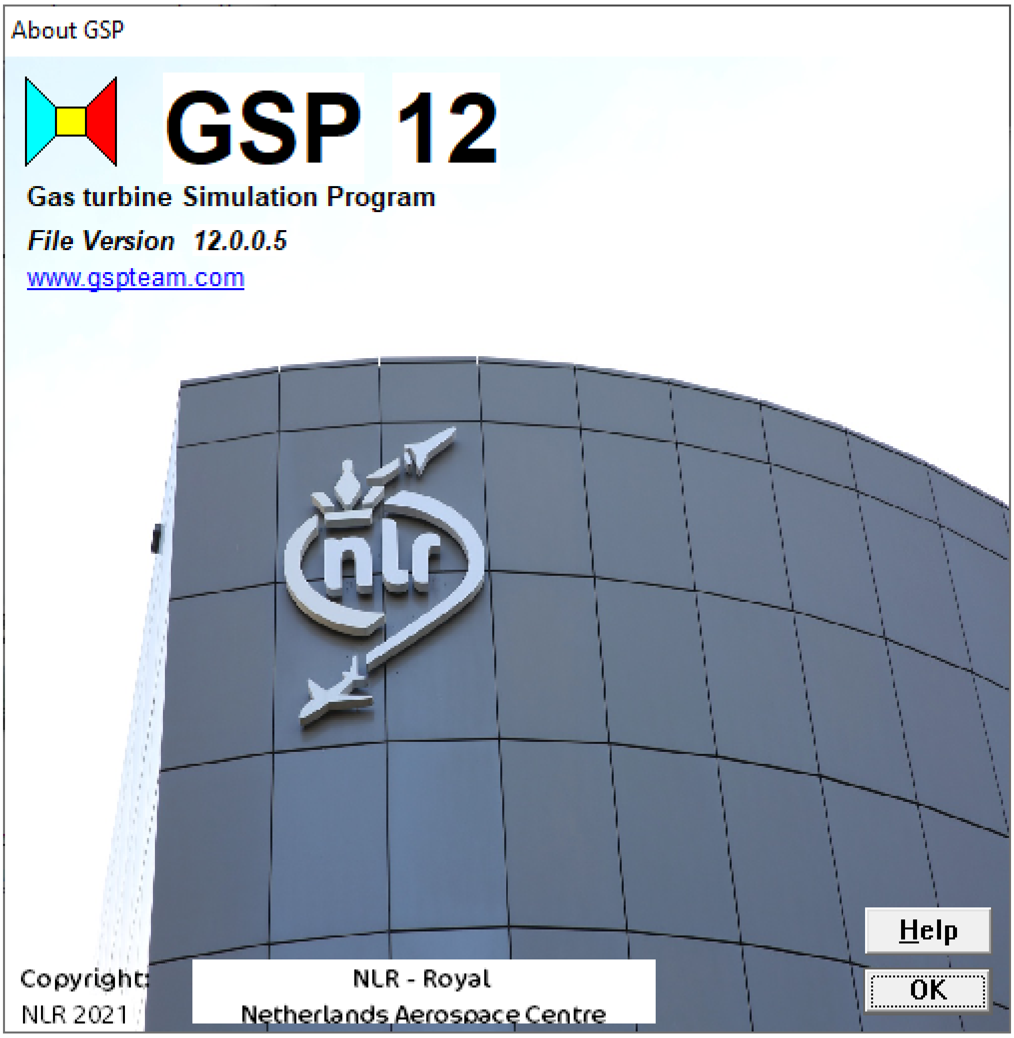
The following components have saving to project file **enabled** in the LE edition (also see the Feature Matrix):

- Inlet
- Compressor
- Combustor (also afterburner)
- Turbine
- Exhaust Nozzle
- Duct
- Fan
- Equation Scheduler
- Constant Expressions
- Manual Fuel control
- Manual Variable Exhaust Nozzle control
- Power Turbine load control
- Compressor Bleed control
- Thrust Control
- Rotor Speed Control

All other components are fully functional, but saving of user defined settings is disabled, this implies that the model components need to be edited manually after loading the model because the default input values are inserted by default by GSP.

4.1.3 About

The about window provides information on copyright and product version. The `File Version` shows the complete version number string of the main program executable. If the GSP installation is a Light Edition, this is shown in red in the middle of the About box. Clicking www.gspteam.com and www.nlr.nl takes you to the associated web sites.



4.1.4 Component Developers Package

The GSP Component Developers Package (CDP) license enables the licensee to be able to develop their own component models or extend simulation capabilities by making new component models inheriting from existing components. The component model source code (Delphi, object pascal) is provided for adaptation or extending component model properties using the object oriented inheritance mechanism of GSP's component model architecture. The resulting custom component libraries can run with the standard licence GSP main program. For using the CDP, the GSP Technical Manual and the GSP Component Model Developer's Manual will be required.

4.1.5 GSP API

The application programming interface (API) specifies how some software components should interact with each other. The GSP API contains functions to control and run models from other programming environments or programs such as MATLAB and SIMULINK. Basically, a GSP model can run embedded is other programming languages and programs using a Microsoft Windows DLL.

4.1.5.1 Microsoft Windows DLL

Dynamic-link library (also written unhyphenated), or DLL, is Microsoft's implementation of the shared library concept in the Microsoft Windows and OS/2 operating systems. These libraries usually have the file extension DLL, OCX (for libraries containing ActiveX controls), or DRV (for legacy system drivers). The file formats for DLLs are the same as for Windows EXE files – that is, Portable Executable (PE) for 32-bit and 64-bit Windows, and New Executable (NE) for 16-bit Windows. As with EXEs, DLLs can contain code, data, and resources, in any combination.

DLL features:

- Since DLLs are essentially the same as EXEs, the choice of which to produce as part of the linking process is for clarity, since it is possible to export functions and data from either.
- It is not possible to directly execute a DLL, since it requires an EXE for the operating system to load it through an entry point, hence the existence of utilities like RUNDLL.EXE or RUNDLL32.EXE which provide the entry point and minimal framework for DLLs that contain enough functionality to execute without much support.
- DLLs provide a mechanism for shared code and data, allowing a developer of shared code/data to upgrade functionality without requiring applications to be re-linked or re-compiled. From the application development point of view Windows and OS/2 can be thought of as a collection of DLLs that are upgraded, allowing applications for one version of the OS to work in a later one, provided that the OS vendor has ensured that the interfaces and functionality are compatible.
- DLLs execute in the memory space of the calling process and with the same access permissions which means there is little overhead in their use but also that there is no protection for the calling EXE if the DLL has any sort of bug.

4.1.6 Feature Matrix

date:	19-09-2022	✔ = *Fully functional* **no save** = *fully functional but changes cannot be saved to project file (LE Edition)* **source code** = *source code available with CDP Component Developers Package license for user adaptation/extension*			
build:	12.0.1.2				
Library	**Component Name**	**Remark**	**GSP 12 LE**	**GSP 12 Standard**	**GSP 12 CDP**
Gas Path	Inlet		✔	✔	source code
	Compressor		✔	✔	source code
	Combustor (or afterburner)		✔	✔	source code
	Turbine (or power turbine)		✔	✔	source code
	Exhaust Nozzle (Con or Con-Di)		✔	✔	source code
	Duct		✔	✔	source code

	Link-bar	Links controls or gas path components	✔	✔	source code
Controls	Fuel Flow control	Manual fuel input	✔	✔	source code
	Manual Variable Exhaust Nozzle control		✔	✔	source code
	Power Turbine Load control		✔	✔	source code
	Compressor Bleed control		✔	✔	source code
	Bleed Schedule Control		no save	✔	source code
	Compressor Shaft Speed Governor		no save	✔	source code
	Turboshaft Governor Fuel control		no save	✔	source code
	Flow Splitter Control		no save	✔	source code
	Inter Cooler Control		no save	✔	source code
	Back Pressure Exhaust Control	Control the ambient static pressure behind the exhaust	no save	✔	source code
	Pressure Vessel Control	Control the pressure and temperature leaving the vessel	no save	✔	source code
	Fuel Mixer	Mix 2 types of custom composition fuels	no save	✔	source code
	Variable Exhaust Nozzle Control (PLA controlled)		no save	✔	source code
	Propeller Control		no save	✔	source code
	Variable Geometry Control	Control IGV, VSV variable geometry	no save	✔	source code
	Property Control	Override component properties	no save	✔	source code

Case control	Operating Envelope Scheduler	Flight envelope scheduler	no save	✔	source code
	Manual Case Control	Manual case input scheduler	no save	✔	source code
	Loop Case Control	Multiple (3) loop case input scheduler	no save	✔	source code
	Monte Carlo Case Control	Monte Carlo case input generator	no save	✔	source code
Power Control	Thrust Control		✔	✔	source code
	Rotor Speed Control		✔	✔	source code
	Power Controller		no save	✔	source code
	Afterburner Control		no save	✔	source code
	EPR Control	Engine pressure ratio controller	✔	✔	source code
Multi In/Out	Fan		✔	✔	source code
	Mixer		no save	✔	source code
	Flow Splitter		no save	✔	source code
	Heat Exchanger/Recuperator		no save	✔	source code
Gas Path Special	Multi Reactor Combustor	Define combustor by section	no save	✔	source code
	Turbine Stage	Single turbine stage component	no save	✔	source code
	Back Pressure Exhaust	Specify alternative ambient static pressure behind the exhaust	no save	✔	source code
	Inter Cooler		no save	✔	source code
	Custom Composition Inlet	Define a user specific composition for the flow	no save	✔	source code
	Pressure Vessel	Constant volume pressure vessel	no save	✔	source code

	Rotating Duct		no save	✔	source code
	Fuel pre-Mixer	Pre-mix fuel to the gas path	no save	✔	source code
Scheduling	1-D Lookup Table Scheduler	Adds user specific equation to the engine model using a table relation	no save	✔	source code
	2-D Map Scheduler	Adds user specific equation to the engine model using a map relation	no save	✔	source code
	Equation Scheduler	Adds user specific equation to the engine model	✔	✔	source code
	Generic Schedule Control	Combines functionality of 1-D, 2-D and Equation scheduler	no save	✔	source code
	Limiter	Parameter limiter	no save	✔	source code
	Design Point Equation Control	Adds user specific design equation to the engine model	no save	✔	source code
Auxiliary	Heat Sink	Define component heat loss	no save	✔	source code
	Propeller		no save	✔	source code
	Installation Effects	Correct to installed performance	no save	✔	source code
	Constant Expressions	Adds the ability to use scaling factor in all numeric input fields	✔	✔	source code
	Additional Output Parameter	Adds user defined output parameters for use in equations or to output table	no save	✔	source code
	Transfer function	Transform output parameter based on selected function, very suitable for sensor dynamics	no save	✔	source code

	Sticky Note	Add a sticky note to the model window	✔	✔	source code

4.2 System requirements

GSP 12 runs on MS-Windows operating systems with the following configurations:

	Recommended
Operating System	Windows 10
Processor	Intel or AMD 2 GHz processor or faster
RAM Memory	1 GB
Resolution	SVGA 1280x1024 or higher @ 256+ colors

4.3 History

Development of GSP was initiated at the Aerospace Department of the Delft Technical University TUD in 1986 where for jet- and turbofan engine simulation NASA's DYNGEN program (Ref. 2) was used. Since DYNGEN had a rather poor user interface and problems with numerical stability, the first version of GSP was developed by Wout Bouwmans, inheriting features from DYNGEN combined with improved stability and speed of the numerical iteration processes.

After 1989 development continued at NLR, where GSP was converted to standard ANSI FORTRAN-5 and implemented on a powerful mainframe computer. Further improvements, adjustments and extensions to the GSP program were made to make useful simulation of jet engines possible: processes in the gasturbine were modelled with more detail improving accuracy, a fan model calculating the separation into duct- and core air flow was developed, a power turbine module enabling turboshaft engine simulation was implemented, a heat exchanger component for modelling turboshaft engines with recuperators was realized, the user interface was improved and a number of output presentation features were added.

In 1996 it was decided to implement GSP in the object-oriented Borland Delphi environment, providing enhanced extensibility, maintainability and an even better user interface. The object orientation features were fully applied, with the result that new gas turbine components can be easily derived from existing components, using inheritance. The object-oriented Delphi language offers a strict set of rules guarding well against many bugs.

At the end of 1998 the first version 8.0 was released which includes a totally revised gas model, fully describing gas composition, real-gas and dissociation effects. The new multi-reactor 1-dimensional combustor model is implemented, enabling detailed analysis of emission formation. Also detailed fuel specification options (including fuel composition) have been added, allowing specification of any type of fuel including alternative (bio-mass) fuels.

Implemented on Windows, GSP now has a very user-friendly drag&drop interface, allowing quick implementation of new engine models and quick analysis of complex problems. When new simulation challenges emerge, the Delphi object-oriented setup enables short development times for new components and features.

4.4 Engines simulated

To date, GSP has been used for simulation of a large number of gas turbine engines including:

Turbojets
- GE J85-GE-15
- RR Avon

Turbofans
- PW JT15D
- RR TAY 620
- JT15D4
- CFM-56
- CF6-50
- CF6-80
- PW4056
- AE3007

Afterburning Turbofans
- F100-PW-200
- F100-PW-220
- F100-PW-229
- F135
- JSF PW119 (STOVL afterburning turbofan with liftfan and clutch)

Turboshafts
- Allison 250-C20B
- T55
- T700
- T800
- Makila
- RTM322
- Rover
- Typhoon
- RR Gem 40/42

Industrial Turboshafts
- GT10
- LM2500
- MTT Mk4 (micro turbine)
- OPRA (recuperated turboshaft)
- MS9001FA (heavy duty 'frame 9')
- Siemens V64.3 (heavy duty)

Turboprops
- PW100 family

Load compressor
- Turblow (air compressor / gas turbine),
- Bio-mass gasifiers integrated with industrial gas turbine.

4.5 Modelling Approach

Gasturbine simulation with GSP is based on non-dimensional (or zero-dimensional) modelling of the processes in the different gasturbine components with aero-thermodynamic relations and steady state characteristics ("component maps"). With zero-dimensional modelling, air and gas properties, thermodynamically averaged over the flow cross-areas (in and exit per component module), are used in the calculations.
A gas turbine model is created by arranging ("stacking") different predefined components (like inlet, compressors, combustors, turbines and exhaust nozzles) in a configuration corresponding to the specific gasturbine type to be simulated. The exit gas condition of a component then forms the inlet gas condition of the next component in the configuration.

The processes in gasturbine components are determined by relations among 2 up to 5 parameters defined by component maps and thermodynamic equations. These parameters are air or gas properties and other parameters such as rotor speeds and efficiencies determining the component operating point.

GSP is an "off-design" model. A predefined design point (it should be called reference point, as it is a point that is used for reference for the off-design analyses, it is not necessarily an actual design point; usually the take-off or cruise is used for reference) is calculated first from a set of user specified design point data. The deviation from the design point is calculated by solving a set of non-linear differential equations. The equations are determined by the mass balance, the heat balance, the equation for conservation of momentum and the power (energy) balance for all components.

In case of a transient simulation, the differential equations include time-derivatives. Then, in each time step, dynamic effects are calculated and the solution represents a quasi-steady state operating point.

4.6 Software implementation

The principles of object oriented design/programming (OOD/OOP): *encapsulation*, *inheritance* and *polymorphism* are extensively used in the GSP 8 & 9 modelling engine to enhance code readability, maintainability and upgradability and minimize the occurence of multiple instances of the same code (as was still the case with version 6.0 in Fortran77).

Encapsulation enhances code maintainability and readability by concentrating all data declarations and procedures (both for interface and simulation calculations) in a single code unit.

Inheritance is used to concentrate code common to multiple component types in *abstract* component classes, preventing code duplication and enhancing code maintainability. For example, the abstract 'turbomachinery component class' represents an *abstract* ancestor incorporating all functionality common to compressors, fans and turbines.

Polymorphism is the ability of parameters to represent different object classes and is extensively applied in GSP. For example, the system model code has an abstract (*polymorph*) identifier able to represent any component in the model. During simulation, the abstract identifier subsequently represents all components and runs their simulation codes.

GSP's graphical user interface fully reflects the object-oriented architecture for the gas turbine system and component models.

The approach of object-orientation enables NLR to derive new or specific application dedicated components very rapidly from existing ones (using inheritance).

4.7 Where to find Information

The following source of information on GSP are available:

- *Online Help*
  Context sensitive Online Help is available while running the program. Referring to Help is an efficient way to learn about the simulation environment and component model features. The 'Quick start basics' tutorial section is an efficient way to quickly learn the GSP basics.
- *Printed documentation*
  Printed documentation is provided in the *GSP User Manual* and *GSP Technical Manual.* The User Manual primarily provides information necessary to use the program for running simulations on existing gas turbine models. The GSP Technical Manual provides detailed information on the thermodynamics and numerical mathematics applied in the simulation environment and component models. This manual is required for more advanced use, including the development of new gas turbine models and new component models. You can download documentation from the GSP site or contact NLR.
- *Internet homepage*
  The GSP website can be found at www.gspteam.com. The web site provides the latest news, product information, downloads of the standard version and all available documentation, tips & tricks, FAQ's, and more. It should be the place to visit in case of obtaining any information or providing feedback.
- *NLR support*

4.8 Documentation

Public printed documentation is provided in:

- *GSP User Manual*
  The User Manual primarily provides information necessary to use the program for running simulations on existing gas turbine models. The User Manual is the printed version of the on-line help.
- *GSP Technical Manual* (registered users only)
  The GSP Technical Manual provides detailed information on the thermodynamics and numerical mathematics applied in the simulation environment and component models. This manual is required for more advanced use, including the development of new gas turbine models and new component models.
- *Several publications presented at conferences* (listed in References)

For the Component Developers Package (CDP) license there further is the

- *GSP Component Model Developer's Manual*
  The GSP Component Model Developers Manual is for developing new custom component models and requires the Borland Delphi software development environment and the GSP Component Developers Package

Additional documentation not publicly available includes:

- *Software Requirements Specification (SRS) for GSP*
- *GSP Analysis and Design Document*

Documentation can be obtained from the NLR GSP site. For additional information contact NLR.

4.9 What's new?

```
NLR GAS TURBINE SIMULATION PROGRAM GSP for MS-WINDOWS              RELEASE NOTES
--------------------------------------------------------------------------------
CONTENTS:
---------------------------------
1. Version history
2. Hints
3. Known issues
4. Other
---------------------------------


********************************************************************************
1. VERSION HISTORY
********************************************************************************
GSP v12.0.4.0                                                         17-03-2023
--------------------------------------------------------------------------------
* Release


GSP v12.0.3.2                                                         16-03-2023
--------------------------------------------------------------------------------
Improvements
--------------------------------------------------------------------------------
* Unit system change improvements (O. Kogenhop, NLR).


Fixed
--------------------------------------------------------------------------------
* Bug in external component set to correctly display the unit strings of numeric
  input fields upon changing unit system (O. Kogenhop, NLR).


GSP v12.0.3.1                                                         15-03-2023
--------------------------------------------------------------------------------
* Customer release


GSP v12.0.3.0                                                         22-02-2023
--------------------------------------------------------------------------------
* Release


Improvements
--------------------------------------------------------------------------------
* If sample maps and projects are not in the user documents, the user is asked
  to copy the default sample maps and projects to the user documents folder when
  browsing for a project (File -> Open Project) (O. Kogenhop, NLR).


Fixed
--------------------------------------------------------------------------------
* Missing GSP12.FDB file in user profile Roaming folder (NLR/GSP) due to company
  scripted installation of GSP installer file -> non-existing directory and file
  will now be created if not existing.


GSP v12.0.2.2                                                         14-12-2023
--------------------------------------------------------------------------------
Fixed
--------------------------------------------------------------------------------
* Copying of a tree node with underlying config or case nodes will now have
  unique table names, not the source table names (O. Kogenhop, NLR).
* Missing GSP12.FDB file in user profile Roaming folder (NLR/GSP) due to company
  scripted installation of GSP installer file -> non-existing directory and file
  will now be created if not existing.


GSP v12.0.2.1                                                         12-12-2022
--------------------------------------------------------------------------------
* Release


Improvements
--------------------------------------------------------------------------------
* Additional API function to get the unit of an output parameter


Fixed
--------------------------------------------------------------------------------
* 12.0.2.0 installer problem with NativevXml.bpl


GSP v12.0.2.0                                                        29-11-2022
--------------------------------------------------------------------------------
* Release


Fixed
--------------------------------------------------------------------------------
* Pre-12.0.1.4 created tables in GSP 12 are now properly converted to the new
  tables. The user can decide if the table needs to be copied or not (if not the
  user is asked whether to delete the old table or not). (O. Kogenhop, NLR).


GSP v12.0.1.5                                                        22-11-2022
--------------------------------------------------------------------------------
Improvements
--------------------------------------------------------------------------------
* Node depth is now an option, default, node depth is limited (as of 12.0.1.4)
  but is now an option for those that prefer model depth more important than
```

```
  model loading performance (O. Kogenhop, NLR).
* Table names aren't constructed from the user defined name or tree node
  location, instead a unique integer based on milliseconds after a certain date
  is used (prefixed with a "T") (O. Kogenhop, NLR).
* Tablename / ID is shown on the statusbar of the table window, in the lower
  right corner for accessing the database manually with e.g. FlameRobin FireBird
  database management tool for offline use (O. Kogenhop, NLR).
* Limited configuration/case name default introduced in 12.0.1.4 can now be
  overridden in the environment options like the node depth option (O. Kogenhop,
  NLR).

GSP v12.0.1.4                                                       02-11-2022
-------------------------------------------------------------------------------
Improvements
-------------------------------------------------------------------------------
* Firebird 3.0 case tablenames are now stored in the XML case nodes as atribute
  named 'TABLE'. This means the table names remain fixed in case the case name
  or location in the tree changes. The initial node index and case names are
  used to compose the table name.
  (Wilfried Visser)
* Now Case nodes can be moved up, down or (copied) to other level without
  loosing the connection to the output table with result records.
  (Wilfried Visser)
* Limits to the configuration and case tree level depths, to avoid slow response
  potential table and trigger naming problems.
  (Wilfried Visser).
* ModelParser now created runtime (no design time component anymore).
  (Wilfried Visser)
* Break- and Groupbreak rows in output table now colored light and dark grey for
  cleares presenation of series between breaks and groups.
  (Wilfried Visser)

GSP v12.0.1.3                                                       14-10-2022
-------------------------------------------------------------------------------
Improvements
-------------------------------------------------------------------------------
* Implementation of Velocity coefficient Cv in Convergent nozzle: Cv does not
  affect choking PR anymore (should not, causes inconsistent results, slightly
  higher Wc (jsut before choking point) than choked flow). Only
  resulting throat exit velocity and static temperature are affected.
  Exit static pressure not affected by Cv (see ConvDuctcode in GSPglobal; for
  Con-Di nozzle a similar approach was already in place).
  (Wilfried Visser).

Fixed
-------------------------------------------------------------------------------
* Problem with error when clearing table and multiple cases selected: now error
  message shown (Wilfried Visser).

GSP v12.0.1.2                                                       16-09-2022
-------------------------------------------------------------------------------
Improvements
-------------------------------------------------------------------------------
* Sample models revisited and updated (O. Kogenhop, NLR).
* Fixed Power Turbine governor controller failing to load component model data
  (O. Kogenhop, NLR).

New features
-------------------------------------------------------------------------------
*

Fixed
-------------------------------------------------------------------------------
* Loading model notes textual stored linebreaks converted to linebreaks
  (O. Kogenhop, NLR).


GSP v12.0.1.1                                                       06-09-2022
-------------------------------------------------------------------------------
Improvements
-------------------------------------------------------------------------------
* API enhancements (SQL tables/storage are/is not used by API, code excluded
  (O. Kogenhop, NLR).

New features
-------------------------------------------------------------------------------
* Addition af extra alternative fuels; Ammonio, Ethanol and Methanol
  (O. Kogenhop, NLR).

Fixed
-------------------------------------------------------------------------------
* Error message after adding configuration fixed, error message had no impact on
  the program (O. Kogenhop, NLR).
* Plotting variable names with special characters in the name, e.g. "N%1" in the
  output graph (O. Kogenhop, NLR).
* Redrawing output graph on maximizing window (O. Kogenhop, NLR).

GSP v12.0.1.0                                                       14-04-2022
-------------------------------------------------------------------------------
First commercial release of GSP 12
-------------------------------------------------------------------------------
What's new? Highlights (for more details scroll down to the build entries)
-------------------------------------------------------------------------------
* Embedded SQL based database
* Advanced data output table:
  - Powerful options for filtering and sorting output data
  - Comparing of different case output datasets
  - Parameter grouping per model component
  - Coloring of the output data table background per model component
  - Default colors are already set, user can edit and revert
  -
* Implementation of interface and options to output according to standards
  SAE AS755 (station designation and nomenclature) and SAE ARP5571 (performance
```

```
    presentation and nomenclature).
    * Set alternative station string:
      Project window: Options -> Output... -> tab "Output Standards"
    * View in model component layout:
      Project Window:  Options -> General... -> tab "Model", option "Show
      alternative station string in model window"
  * The root node that was called Reference model in GSP 11 is now called
    "Base configuration".


  GSP v12.0.0.12                                                  13-04-2022
  ----------------------------------------------------------------------------
  Improvements
  ----------------------------------------------------------------------------
  * Corrected massflow for turbomachinery components now always set to Wcgas (not
    Wc) because any liquid fractions must be excluded. (W. Visser, VisserTek)
  * Speed of rendering the data table into the graphs has been increased and
    improved. (W. Visser, VisserTek)


    GSP v12.0.0.11                                                07-04-2022
  ----------------------------------------------------------------------------
  Improvements
  ----------------------------------------------------------------------------
  * Faster renumbering of curves and groups in output table (O. Kogenhop, NLR).


  New features
  ----------------------------------------------------------------------------
  * Export the table output to an Excel file (WYSIWYG, so including component
    output background coloring and column component type grouping) (O. Kogenhop,
    NLR).


  GSP v12.0.0.10                                                  10-03-2022
  ----------------------------------------------------------------------------
  Fixed
  ----------------------------------------------------------------------------
  * Problem refreshing carpet and legend field lists fixed. (W. Visser, VisserTek)
  * API: problems compatibility with dx and cx controls fixed (cx controls
    initialized explicitly in Initialization section of form units). (W. Visser,
    VisserTek)


  Improvements
  ----------------------------------------------------------------------------
  * GSP 12 API speed; tested to be at least 30 times faster than GSP 11 API
    (O. Kogenhop, NLR).
  * Ther term "ReferenceModel" or "Reference Model" as root node for the model
    configurations has been dropped in favor of the naming "Base configuration",
    this is the root node from which all other model configurations and cases in-
    herit (W. Visser, VisserTek / O. Kogenhop, NLR).


  New features
  ----------------------------------------------------------------------------
  * GSP API functions added (O. Kogenhop, NLR).


  GSP v12.0.0.9                                                   09-02-2022
  ----------------------------------------------------------------------------
  Fixed
  ----------------------------------------------------------------------------
  * Bugs in initial code for automatic Firebird table recreation when database
    table changes counter approached limit of 255 (W. Visser, VisserTek).
  * RNI (Reynolds index number) is calculated on constant axial Mach number
    instead of constant velocity (W. Visser, VisserTek / O. Kogenhop, NLR).
  * Component envelope scheduler:
    1) Drawing outside flight graph window fixed.
    2) Power control setup can now be done in case model StSt. series, this used
       to be configurable in design mode only.
    (O. Kogenhop, NLR)
  * NaN values not allowed in calculated columns (present in the break rows, this
    caused poblems with drawing the output graph) (O. Kogenhop, NLR).


  Improvements
  ----------------------------------------------------------------------------
  * Automatic Firebird table recreation when database table changes counter
    approached limit of 255 (W. Visser, VisserTek).


  New features
  ----------------------------------------------------------------------------
  * Output table curve counting and renumbering (O. Kogenhop NLR).
  * Added option to add curve and group number into the graph using the graph
    symbol. Use '$' as curve/point symbol in the graph options. (O. Kogenhop NLR).
  * Additional options to number curves through the graph settings "line styles"
    (O. Kogenhop NLR).


  GSP v12.0.0.8                                                   05-01-2022
  ----------------------------------------------------------------------------
  Fixed
  ----------------------------------------------------------------------------
  * Internal table naming based on case naming is now truncated (for users there
    is no change, users can still use long configuration and case model names).
    (O. Kogenhop NLR)
  * Fixed table parameter redefinition when more than 255 changes are scheduled,
    this is to circumvent a feature of the currently used DB. (W.Visser VisserTek)
  * Project window security options linked to new SQL database tables from the
    GSP main window "Environment options". (O. Kogenhop NLR)
  * Option to clear the security markings. (O. Kogenhop NLR)
  * Specific heat calculation in turbine fixed. (W.Visser VisserTek)
  * Deleting of breaks from table view. (O.Kogenhop NLR)


  Improvements
  ----------------------------------------------------------------------------
  * Enforcing AS755/ARP5571 parameter naming is done via a separate option,
    default GSP parameter naming prevails unless the users check the "Rename
    parameters to recommended" output option. (O. Kogenhop NLR)
  * Improved the station renumbering to AS755 recommended station strings.
    (O. Kogenhop NLR)
```

```
* Case management is now enforced on all projects, the old model .MDL support is
  now discontinued. Pre GSP 11 models must now all have been converted.
  (O. Kogenhop NLR)
* Various updates and improvements of the GUI and hel manual. (O. Kogenhop NLR)


New features
--------------------------------------------------------------------------------
* Possibility to show alternative station string (3-digit numbering) in the
  model window. Showing the alternative station is controlled through the
  Project Window:  Options -> General... -> tab "Model", option "Show
  alternative station string in model window". (O. Kogenhop NLR)


GSP v12.0.0.7                                                         27-10-2021
--------------------------------------------------------------------------------
Fixed
--------------------------------------------------------------------------------
* Problem skipping trasnient/steady-state series output points.
  (W.Visser VisserTek)
* Problem output parameter selection comboboxes in control components not filled
  fixed. (W.Visser VisserTek)
* Output table <auto-break> bug fixed; a break sometimes was added after the
  first series results record if a break prior to the record was present (e.g.
  as break in between DP and first OD results record, project window Options ->
  Output... -> Output Tables option  "Add break between Init. Design and St.St."
  is checked). (O. Kogenhop NLR)


Improvements
--------------------------------------------------------------------------------
* Color picker for Ambient/Flight conditions window added. Color is reverted on
  cancelling window if changed. Added popup menu to revert to default color.
  (O. Kogenhop NLR)
* Added default output colors for all model components, color is reverted on
  cancelling window if changed. Added popup menu to revert to default color.
  (O. Kogenhop NLR)
* GUI improvements of window controls. (O. Kogenhop NLR)


GSP v12.0.0.6                                                         08-10-2021
--------------------------------------------------------------------------------
Fixed
--------------------------------------------------------------------------------
* Problem progressbar not updating during project load (W.Visser VisserTek).
* Problem with divide by zero when using DP equation schedulers.
  (W.Visser VisserTek)


Improvements
--------------------------------------------------------------------------------
* Much faster loading projects from mainform (W.Visser VisserTek).
* GUI improvements of window controls. (O. Kogenhop NLR)


GSP v12.0.0.5                                                         03-10-2021
--------------------------------------------------------------------------------
Improvements
--------------------------------------------------------------------------------
* GUI improvements of window controls. (O. Kogenhop NLR)


GSP v12.0.0.4                                                         29-09-2021
--------------------------------------------------------------------------------
Fixed
--------------------------------------------------------------------------------
* Problem deleting rows with multiple cases selected solved.
  (W.Visser VisserTek)


GSP v12.0.0.0                                                         09-03-2021
--------------------------------------------------------------------------------
New features
--------------------------------------------------------------------------------
* First version 12 release with Firebird 3.0 database. Database per projects
  with tables per run case, saved in database with same name as project. This
  means a projects can consist of 2 files:
  - projectname.mxl
  - projectname.FDB (the simulation result data, optional: if not existing a new
    emptu database will be created).
  Multiple cases can be selected for plotting in the graph panel.
  (W.Visser VisserTek)
```

For a full overview of all release notes, please visit the online manual.

```
********************************************************************************
2. HINTS
********************************************************************************
* Use the Convergence monitor
* Use the 'Global output parameter' options in the Model Output options to
  specify requested output for parameters such as net thrust FN, Thrust specific
  fuel consumption TSFC, fuel flow, shaft power SFC. For calculating total
  turbine shaft power output, use this option instead of Psurplus (used for this
  purpose in previous versions).
* Set printing to Black & White (in Environment options in GSP Main window) when
  coloured curves do not appear.


********************************************************************************
3. KNOWN ISSUES
********************************************************************************
* With some printer drivers, coloured curves do not show: see Hints.
* With the new video system requirement of 64K colors for version 8.2.0.3 and
  up, Help file images are not displayed properly with less than 64K colors.
```

```
*******************************************************************************
4. OTHER
*******************************************************************************
* The new policy of GSP 11.0 and higher has changed with respect to previous
  versions. The 'Light Edition' (LE) used to hold the basic gas path components
  and the basic control components. From version 11.0 all additional components
  (supplementary and STOVL components library) will be accessible in the GSP LE
  versions. Modellers will be able to configure/set up models with these
  components, but running will be prohibited. Running of models containing
  registered components requires a license. This new policy allows potential
  customers to browse through available components to evaluate whether these
  satisfy the modelling needs. If not please contact info@gspteam.com for
  additional questions.
* The 'Multi-reactor' combustor model option has been removed from the standard
  component libraries. It is now part of the GSP Generic Components Library for
  registered users. This is part of the policy to issue GSP 9.0 free with full
  simulation functionality with the standard library but with some modelling
  capabilities omitted.


@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

For questions or comments, please contact:

The GSP Development Team
Aerospace Vehicles, Gas Turbines & Structural Integrity (AVGS)
National Aerospace Laboratory NLR
P.O. Box 90502              Anthony Fokkerweg 2
1006 BM Amsterdam  1059 CM Amsterdam

Email    : info@gspteam.com
Internet : http://www.gspteam.com

@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
```

4.10 FAQ

The most up to date FAQ can be found at the GSP website.

Turboshaft power output in output table does not change in off-design calculation?

How do I insert a value into a component input field?

4.10.1 Turbine shaft power output

Question:
How do I calculate power output of a turboshaft (or turboprop) engine?
(I see PWshaft unchanged in the output table although I change off-design conditions. Also if I change ambient conditions I see SCF remain unchanged.)

Answer:
This problem is due to the settings in the power turbine and (if present) load control components.
Also: make sure you do not use the design point calculation when you actually want to predict off-design performance. Note that if you change ambient conditions for example and re-calculate design point, you actually calculate a design point of a different engine.
Load specification and calculation for a turboshaft engine with free power turbine is a little more complicated than specifying jet engine power setting due to various options to power level for maximum flexibility:

a) Without the load control component, for a power turbine running at a fixed user specified speed, GSP calculates the Power surplus (PWsurplus) as a difference between power taken from the shaft by the External Loads specified in the Turbine component and the mechanical power delivered by the turbine. In the design point PWsurplus=0 by definition and the total design power load specified is used to calculate the turbine design point performance. The PWshaft output field in this case only represents what is specified as loads: not necessarily equal to power delivered. Of course, one can derive actual power delivered as the difference between PWshaft en PWsurplus, but this is not as practical as using option b). Also note that the PWshaft_x (with x for component nr. or string) represents gross turbine power output. PWshaft_x times mechanical efficiency equals net total turbine power output.

b) With the load control, a design load can be specified in the Load control (and none in the turbine), while the off-design loads are all left to 0. Now GSP uses the load control to correctly calculate actual power output from the power loads specified and the surplus power calculated: the PWshaft will always show the correct power output provided the turbine and load control components are correctly configured (see below).

Power levels and torques can be combined (to be calculated as a total load) in the turbine and load controls. It is usually best to use a Loadctrl component (option b)) to efficiently use GSP for turboshaft/turboprop performance calculations.
There are basically two alternative approaches to calculate off-design performance:

1) specify fuel flow and calculate power output:
   You then normally have selected the `Specify fuel flow` and Fuel flow input `User specified` options in the Manual fuel control (or PLA in a detailed fuel control model). In the power turbine component, set Model options to `User specified rotor speed` and check the `Free Power Turbine` check box. Leave all design and off-design loads in the turbine component at value 0, unless a fixed auxiliary power is continuously extracted. Use a Load control component to specify turbine load. Set the right shaft nr./suffix (equal to the power turbine component shaft). Leave the off-design loads (in the General tab sheet) at value 0 and only add a design point load. It is usually better to specify a torque load instead of a power value because with torque, the power will vary with any user specified rotor speed changes accordingly. Torque as input is also more practical if analysing engine transient responses to load variations (torque steps). Make sure you clear the `Auto reset input to design` option in the Model `General` options (Model `Options| General` menu) to avoid the initial setting of off-design load to design load at design point calculation, otherwise you must reset off-design load to 0 after every design point calculation.
2) specify power output and calculate fuel flow (or power setting PLA) required:
   Activate the `Auto reset input to design` option in the Model General options (Model `Options| General` menu) to initially set all off-design input to design at design point calculation. In the Manual fuel control component (or control system model component PLA) change the Fuel flow input option to `Free state` (this adds a 'State variable' to the GSP equation system). In the Power turbine component change the Model Options to `Power balance at rotor speed` (this adds a corresponding 'Error variable' to the GSP equation system). After design point calculation the off-design load in the Load control now is initialized to design values. With subsequent off-design (stead-state or transient) calculations at varying load levels in the Load control General (or transient) tab sheet, now GSP calculates the corresponding fuel flow required.

As an alternative, Load control input can be specified as a state instead of fuel flow, which in effect turns 2) back into option 1). Then the load corresponding to a user specified fuel flow is solved form an extra equation. This option however is not recommended because of potential iteration stability problems with the extra equation in the system.
These guidelines equally apply to single spool turboshaft engines (fixed power turbine). The only difference is that a part of the turbine power delivered is used for driving the compressor.

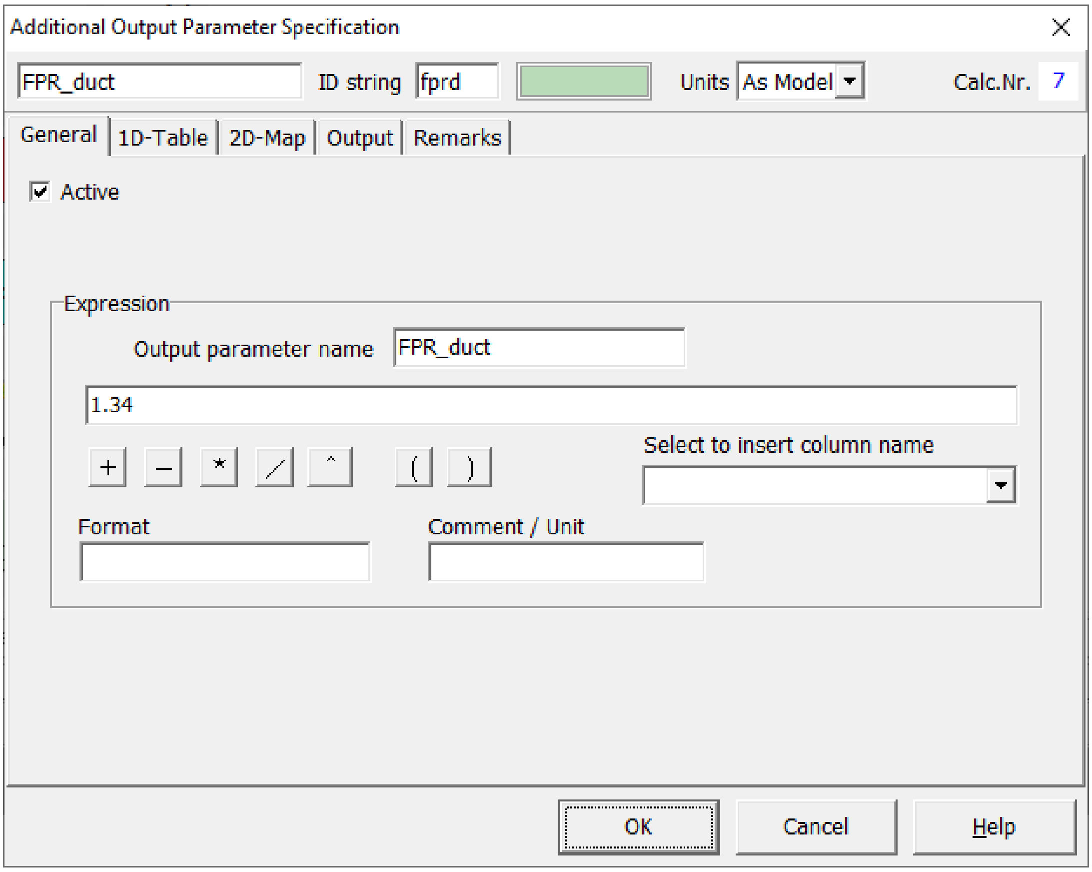
4.10.2 Insert (Off-)Design Value(s) in model component

Sometime you'd like to insert variables into model components to sweep certain parameters that do not have the means default to be swept (there is no steady state series input option).

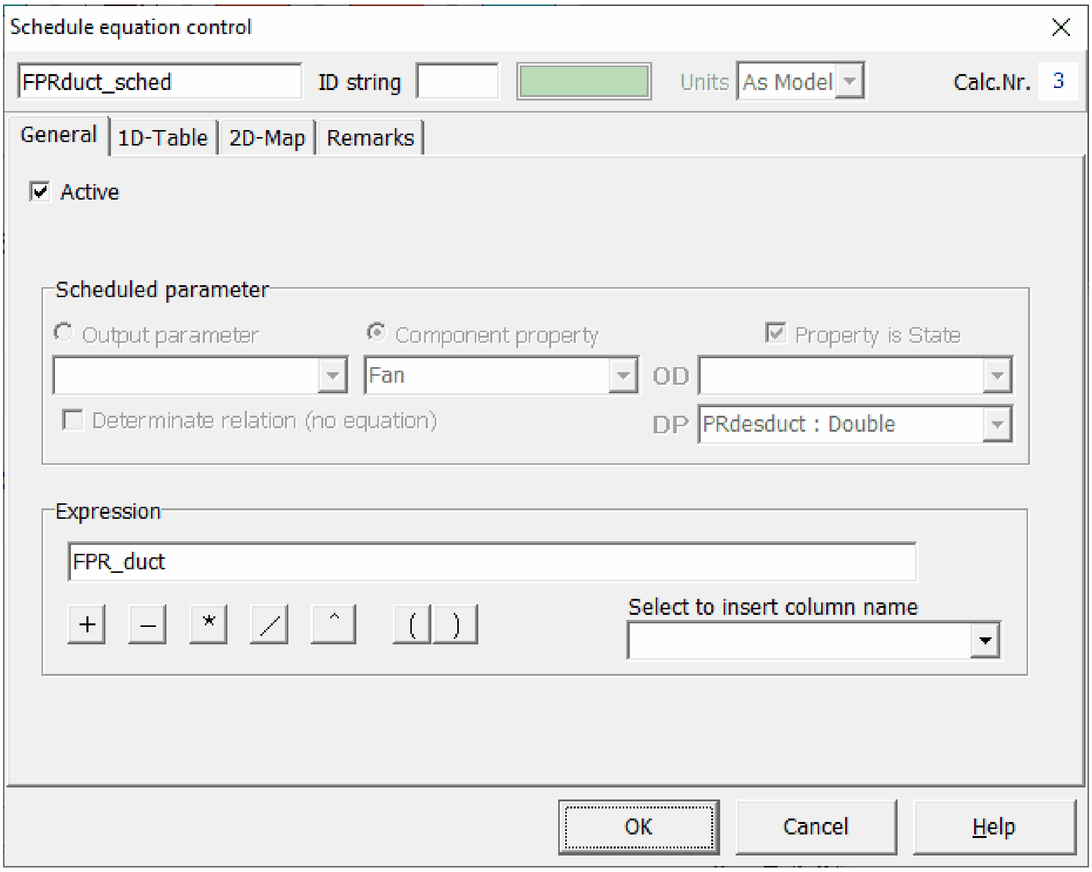
It is possible to insert most steady state (off) design input through the use of other components.

To do so it is required to add an additional output parameter model component, e.g. the following example will demonstrate the use of the duct PR for design sweep analysis.

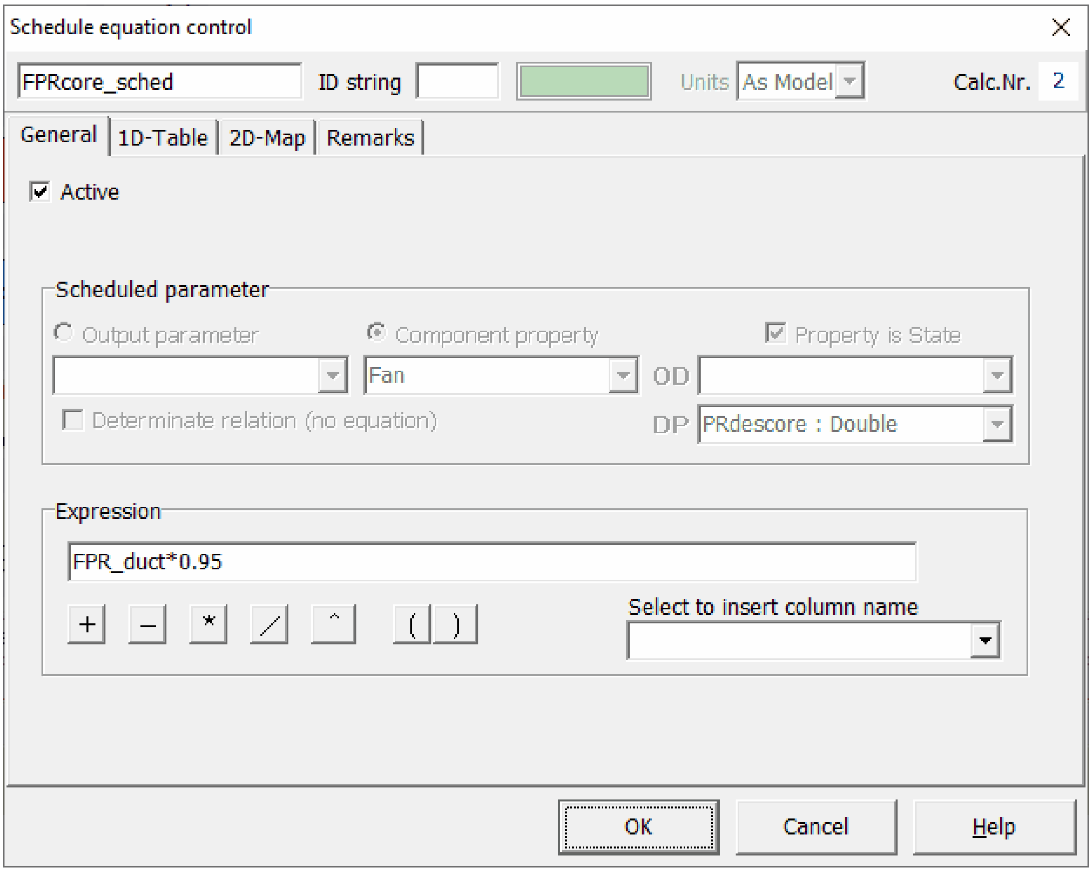
FPR_duct is the name of a parameter we made up, fill out a random value, it will be overwritten later (note to edit the input from a configuration, the screenshots are from a case model, so it appears disabled):



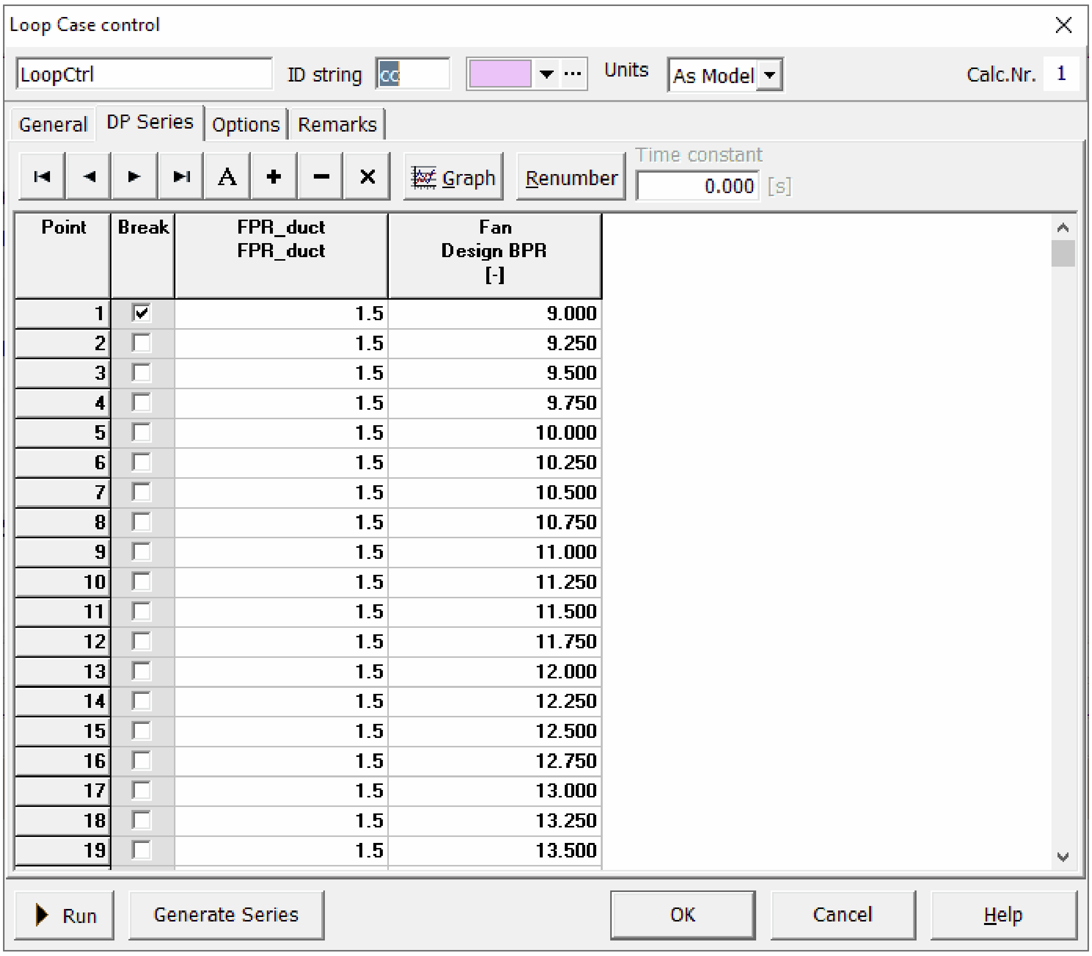
Now we need to link this parameter to the actual input in the fan (fan duct pressure ratio input fileld), to do so we use a equation model component and configure it s such that it links to the design PR of the fan (PRdesduct : Double):



Similar for the fan pressure of the core (PRdescore : Double):



Now the model can schedule the fan pressure design values from a case input controller:



Point	Break	FPR_duct FPR_duct	Fan Design BPR [-]
1	☑	1.5	9.000
2	☐	1.5	9.250
3	☐	1.5	9.500
4	☐	1.5	9.750
5	☐	1.5	10.000
6	☐	1.5	10.250
7	☐	1.5	10.500
8	☐	1.5	10.750
9	☐	1.5	11.000
10	☐	1.5	11.250
11	☐	1.5	11.500
12	☐	1.5	11.750
13	☐	1.5	12.000
14	☐	1.5	12.250
15	☐	1.5	12.500
16	☐	1.5	12.750
17	☐	1.5	13.000
18	☐	1.5	13.250
19	☐	1.5	13.500

A similar solution can be applied for off-design input fields.

4.11 How to...

This section contains descriptions of a number of GSP tasks which usually are performed very frequently. Either use the help index or the contents section to find help on how to perform a specific task not listed below.

Q: How do I calculate a steady state point at a specified gas generator speed?
A: In case of a manual fuel control component: Set the fuel flow as a free state in the component options, set the compressor speed to 'Externally controlled' and set the gas generator turbine model option to "Power balance at rotor speed'. Specify the required turbine rotor speed.
In case of a governor or custom fuel control component model, use the 'Fully trimmed steady state (no droop)' option to have GSP find the fully stabilized steady state operation point for the specific control input (input PLA determined demanded rotor speed).

Q: How do I calculate power and/or torque output of a turboshaft engine?
A: The best way is to add a Load Control component for flexible control of the turbine load, also as a function of time. Activate the Shaft Power PW and SFC check boxes in the `Output`

`Options|Global Output Parameters` tab sheet to get the typical primary turboshaft power output data. Alternatively the Net power/torque output option in the Load control component output tab sheet may be used to obtain power and torque output for the Load Control only (which usually equals total load).

Q: I get unexpected off-design results, results deviating from the design point where they should not. Wat causes this behaviour?
A: A cause may be inadvertent settings of off-design operating conditions such as ambient/flight conditions or deterioration. Make sure all these are set correctly (initially/usually equal to design settings) in all components.
Note that the 'Auto reset input to design' General model option automatically resets all of these back to design during a design calculation. With this option off, all off-design inputs remain, even during design point calculation.
Finally, make sure any inadvertent transient inputs do not require unreasonable operating conditions: deactivate the transient inputs in the transient tabsheets if they have more that a single row but are not to be used.

Q: GSP cannot find a steady state (or transient) operating point (Error message: 'Number of steps exceeded limit of ...'). How do I fix this?
A: Although GSP has a very powerful solver, due to the very non-linear nature of gas turbine characteristics there are cases (complex models/control laws/operating conditions) where GSP cannot find the operating point. The best approach to avoid GSP from getting 'stuck' somwhere is to start calculating points in small steps from the design point onwards, using the steady state series or transient modes. Also manually one can adapt fuel flow for example in small steps starting at the design value.
If the above message appears it is often best to recalculate the design point again and proceed from there with small steps.
The error may also indicate there is no solution at all (for example: with a very low fuel flow, there may not exist a steady state operating point at all).
Another cause may be inadvertent settings of off-design operating conditions such as ambient flight conditions or deterioration. Make sure all these are set correctly (initially/usually equal to design settings) in all components.
Note that the 'Auto reset input to design' General model option automatically resets all of these back to design during a design calculation.
Finally, make sure any inadvertent transient inputs do not require unreasonable operating conditions: deactivate the transient inputs in the transient tabsheets if they have more that a single row but are not to be used.

4.12 Authors

GSP is continuously being developed since 1996 by a steadily growing group of authors and software developers representing the *GSP Development Team* or *"GSP Team" of* www.gspteam.com. The GSP Team includes members from NLR, Delft University of Technology and users/developers from several institutes and industries. The coordination of the development work is performed by NLR and Delft University.
Below, the authors of the main GSP elements are listed (see also References):

(updated December 2013)

Main program

- Kernel, object oriented architecture and design (1996-1998)
  Wilfried P.J. Visser
- Main program, standard component library and GUI
  Wilfried P.J. Visser, Oscar Kogenhop, Michael Broomhead
- Case management architecture and XML storage and inheritance mechanism
  Wilfried P.J. Visser, Oscar Kogenhop

- Generic control system component library
  Wilfried P.J. Visser, Oscar Kogenhop, Michael Broomhead
- Parametric P3T3 emission models in combustor module
  Michiel Bruin
- Gas and combustion model for user specified fuels
  Wilfried P.J. Visser
- Multi-reactor emission models in combustor module
  Steven Kluiters, Wilfried P.J. Visser, Edward R. Rademaker
- Thermal network simulation functionality / heat sink component
  Wilfried P.J. Visser

Component Developers Package

- A special source code release for customer in-house development
  Oscar Kogenhop,
  Wilfried P.J. Visser

Application Programming Interface

- Development of a windows dll file containing functions to simulate GSP models from other software or programming languages
  Oscar Kogenhop,
  Wilfried P.J. Visser
- Development of a MATLAB-SIMULINK S-Function to allow usage of GSP in SIMULINK (C-code development)
  Erik H. Baalbergen,
  Oscar Kogenhop

Application specific libraries

- Additional libraries with:
  - custom control system components
  - custom heat exchanger components
  - custom pressure vessel inlet for turbine powered wind tunnel engine simulator model
  - customer specific components

  Oscar Kogenhop,
  Wilfried P.J. Visser,
  Edward R. Rademaker
- STOVL specific components
  Wilfried P.J. Visser
  Michael Broomhead
- Supplementary components
  Oscar Kogenhop,
  Michael Broomhead,
  Wilfried P.J. Visser
- Gas path analysis / Adaptive modeling library
  Wilfried P.J. Visser
  Oscar Kogenhop
  Mark Oostveen
- TUD / KLM Gas path analysis / Adaptive modeling library
  Wilfried P.J. Visser

Michel Verbist

- MTT micro turbine component library
  Wilfried P.J. Visser

- NLR component library
  Oscar Kogenhop

Documentation

- GSP User Manual / On-line help
  Oscar Kogenhop
  Wilfried P.J. Visser
  Michael Broomhead
- GSP Technical Manual
  Oscar Kogenhop
  Wilfried P.J. Visser
  Michael Broomhead
  Edward R. Rademaker

- GSP Component Developers Package (CDP) Manual
  Oscar Kogenhop
  Michael J. Broomhead
  Michiel J.D. Valens
  Wilfried P.J. Visser

- GSP API Manual
  Oscar Kogenhop
- Heat transfer modeling
  Michel Verbist

5 Quick Start Basics

The quickest way to introduce yourself to the GSP environment is to use this tutorial section which guides you through the basics of GSP using simple examples.

Follow the instructions to

- Learn to create a GSP project file based on a sample model
- Run simulations of a simple straight jet engine to analyse design and off-design performance,
- Analyze the effect of changing engine characteristics,
- Extend the straight jet engine sample model to create an afterburning bypass engine.

5.1 Your first simulation session

Quick start basics .

This tutorial will introduce the basics to use GSP for engine modeling using simple examples excercises.

Note: After completion of the tutorial you have only used a small part of GSP's power. It is advised to use on-line help anywhere you can. For more in depth information read the Technical Manual before starting complex tasks.

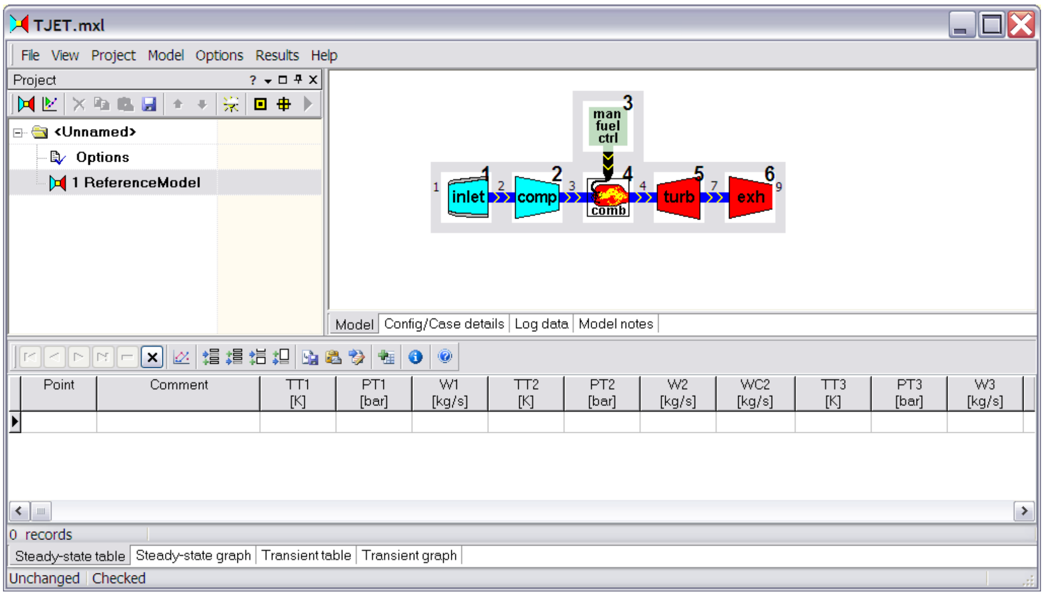
5.1.1 Opening a project

Quick start basics - Your first simulation session .

Before running a simulation, you must start GSP and create a project or open an existing project. A GSP project may contain one or more gas turbine model configurations. Configurations normally will have one or more run cases to perform the actual simulations. Configurations and cases are organized in a tree structure on the left of the project window. There is always a single root configuration named the Base configuration model (in GSP 11 this is called the "Reference Model").

1. Start GSP by selecting `Start|Programs|NLR GSP 12|GSP 12` from the MS-Windows task bar.
   An introductory splash screen appears (can be accompanied by a jet engine aircraft fly-by sound in case of a multimedia PC, which you can switch off in the environment options).
   Next, the main window appears.

2. Open the existing example project by clicking the `Open` button from the `File` menu, select TJET.MXL from the sample projects directory of the GSP main directory and press `Open`.
   The example project is a simple turbojet engine model configuration representing an engine similar to the General Electric J85 (powering e.g. the Northrop F-5).
   The loading process is reported in the progress bar. After loading, the project window appears:



Save the project using `File|Save Project As...` to the user `GSP\models` directory in the My Documents folder (e.g. `C:\Documents and Settings\<username>\My Documents\GSP\models`).

5.1.2 Starting a simulation session

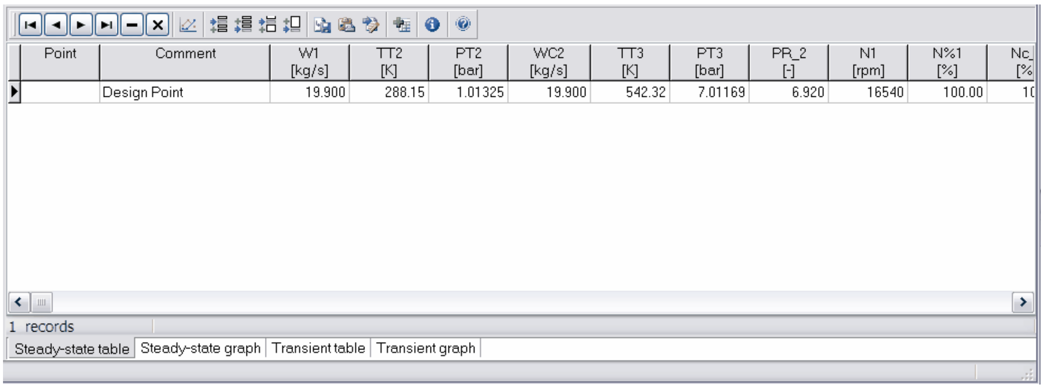
Quick start basics - Your first simulation session .

Once the example project has been loaded a basic configuration has been loaded. For a runnable instance of a model configuration we need to define a case and its case type.

1. Our reference model of the TJET is stored in the Base configuration model (the model root) node. From this base configuration model we will perform some performance analysis. First add a configuration to the Base configuration model configuration. The reason for adding an extra layer is that this might be very useful if the model should be changed later, and therefore not compromise the original reference/base configuration model. Adding a configuration can be done using one of the 3 different options (described here) to edit the project tree view (the most convenient way is to press the GSP logo button of the toolbar just above the project tree). Confirm the base configuration model changes (if any) dialog and enter the configuration name "TJET_Config" in the `Config Name` dialog. Second, add a case to the reference model by using one of the 3 different options (described here) to edit the project tree view. Confirm the configuration model changes (if any) dialog and enter the case model name "Design case" in the `Case Name` dialog and set the case type to design by selecting `Design` from the drop down selection box in the right column next to the case name.

   Press the run button (green arrow) or alternatively press F9 to start the simulation.

   Simulation results will be generated in the Steady State output table, located default in the lower right corner of the project window (see figure below).
   To obtain a reference state for the off-design calculations, the actual gasturbine configuration must be defined and the components must be "sized" with a design point calculation. As usually defined, the model design point is the sea-level standard static rated thrust or power, which usually is the rated maximum take-off thrust.



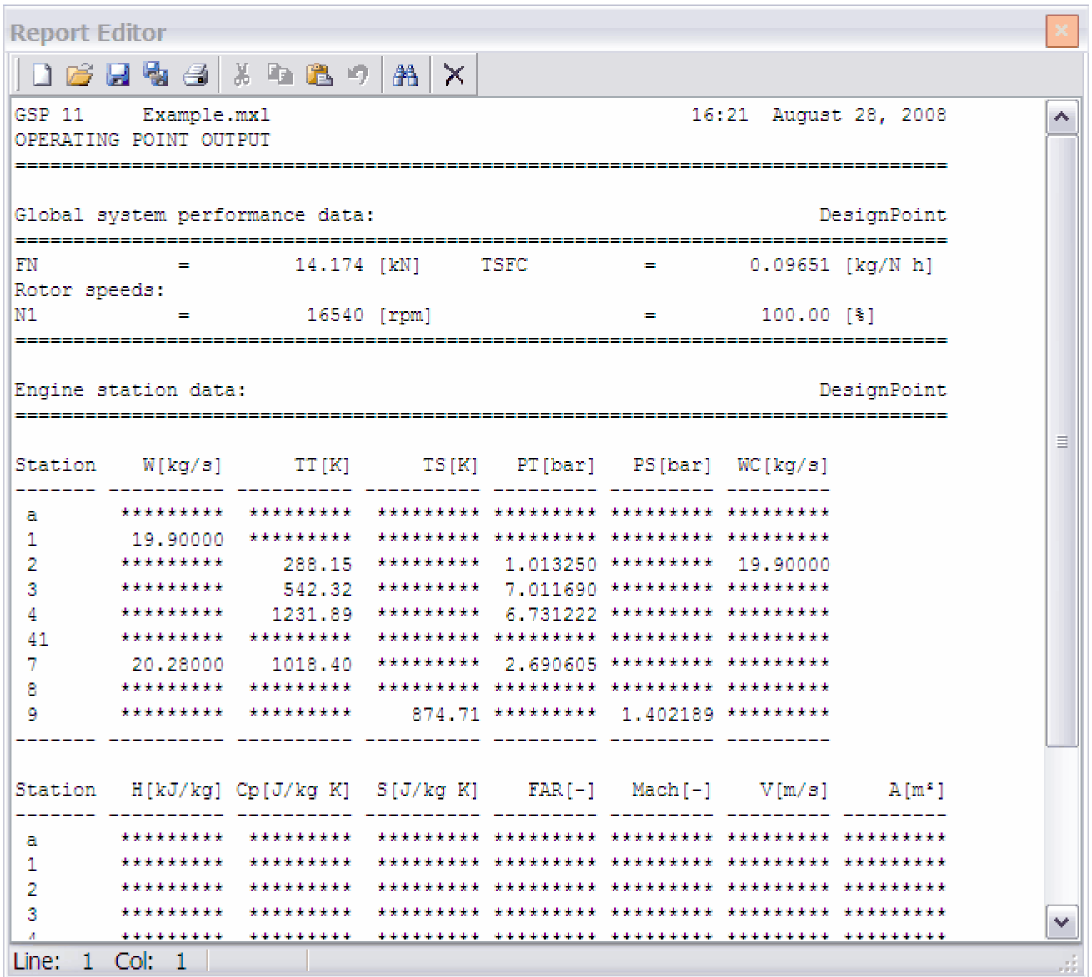
2. Study the values in the table by scrolling left and right using the bottom scroll bar.
The table shows the design point steady state performance data on the first row. On the far right net thrust and SFC can be found. The gas path output parameters shown here have been selected in the respective components' `Output` tab sheet of the data entry window of each component. System performance parameters are set on `Options|Output...|Output parameters|System Performance`.

We have shown here how to calculate the design or the reference point for a gas turbine cycle.

5.1.3 Text output

Quick start basics - Your first simulation session .

To output the design operating point data to a textual report, do the following:
1. Perform a design calculation using the design case model.
2. Click `Results|Operating Point Report` in the model window menu.
The operating point report will be displayed in the report editor (will pop up, dock this window anywhere in the project window). The report displays the data from the last calculated operating point, in this case the design point. The report can be edited and saved for future use in a word processor. If a parameter has not been selected in the respective components' `Output` tab sheet of the data entry window of each component, the string "*********" will be displayed.


Report Editor

```
GSP 11      Example.mxl                                         16:21  August 28, 2008
OPERATING POINT OUTPUT
==================================================================================

Global system performance data:                                         DesignPoint
==================================================================================
FN           =          14.174 [kN]      TSFC          =        0.09651 [kg/N h]
Rotor speeds:
N1           =           16540 [rpm]                   =         100.00 [%]
==================================================================================

Engine station data:                                                    DesignPoint
==================================================================================

Station    W[kg/s]       TT[K]       TS[K]    PT[bar]    PS[bar]   WC[kg/s]
------- ---------- ---------- ---------- --------- --------- ---------
 a       *********  *********  ********* ********* ********* *********
 1        19.90000  *********  ********* ********* ********* *********
 2       *********     288.15  *********  1.013250 *********  19.90000
 3       *********     542.32  *********  7.011690 ********* *********
 4       *********    1231.89  *********  6.731222 ********* *********
 41      *********  *********  ********* ********* ********* *********
 7        20.28000    1018.40  *********  2.690605 ********* *********
 8       *********  *********  ********* ********* ********* *********
 9       *********  *********     874.71 *********  1.402189 *********
------- ---------- ---------- ---------- --------- --------- ---------

Station   H[kJ/kg] Cp[J/kg K]  S[J/kg K]    FAR[-]   Mach[-]    V[m/s]    A[m²]
------- ---------- ---------- ---------- --------- --------- --------- ---------
 a       *********  *********  ********* ********* ********* ********* *********
 1       *********  *********  ********* ********* ********* ********* *********
 2       *********  *********  ********* ********* ********* ********* *********
 3       *********  *********  ********* ********* ********* ********* *********
```

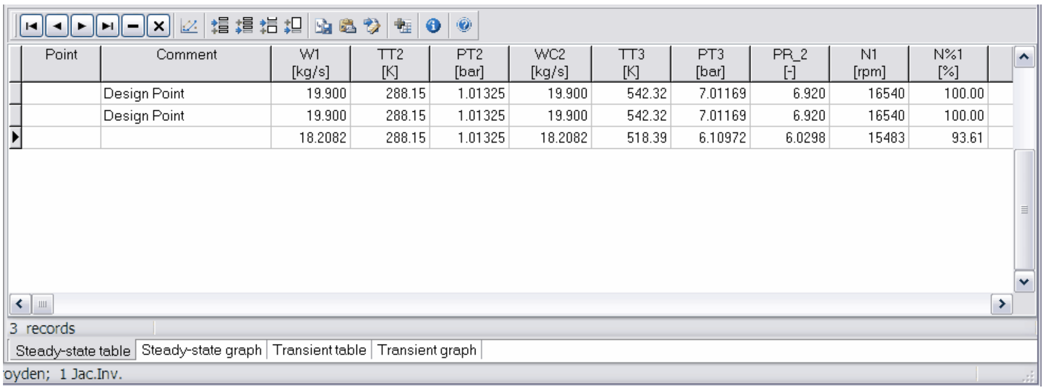
Line: 1 Col: 1

5.1.4 Running steady-state simulations

Quick start basics - Your first simulation session .

With the model initialized to a design point, simulation input in the form of deviating operating conditions can be specified, which will make the engine operate off-design. Model initialization for a design is automatically performed before a steady state calculation as long as the design has not changed or manually the model has been initialized. For a start, let's modify the fuel flow in order to determine a partial power operating point.

1. Click in the Project tree panel on the "TJET_Config" node to load the original configuration. Now add a new case model (name it "Off-design case") and set the case type to `Steady-State`. Double-click the manual fuel control icon of the model panel, or alternately right click on the icon and select `Edit` from the pop-up menu. The component data window for the manual fuel control component is now open.
2. Click the `General` tab.
3. Modify the fuel flow value `Wf` from the design value of 0.38 kg/s to 0.3 kg/s and click `OK`.
4. Press the run button (green arrow) or press F9 to start the off-design steady state simulation. Note that a reference (design) is calculated to scale the engine.
   The table will reappear as shown, showing the off-design steady state performance data corresponding to a fuel flow of 0.3 kg/s on the second row. Study the differences in data.



Point	Comment	W1 [kg/s]	TT2 [K]	PT2 [bar]	WC2 [kg/s]	TT3 [K]	PT3 [bar]	PR_2 [-]	N1 [rpm]	N%1 [%]
	Design Point	19.900	288.15	1.01325	19.900	542.32	7.01169	6.920	16540	100.00
	Design Point	19.900	288.15	1.01325	19.900	542.32	7.01169	6.920	16540	100.00
		18.2082	288.15	1.01325	18.2082	518.39	6.10972	6.0298	15483	93.61

5. Repeat the process for other fuel flows or other simulation input changes.

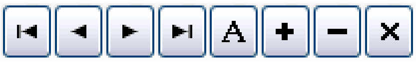
   For example, click the ambient conditions button above the project tree (), increase the altitude value `Zp` on the `Off-design` tab, click `OK` and run the simulation again (F9).
6. Click `Model|Reset OD input to DP values` to reset the model variables and undo changes to load last design values.

5.1.5 Running steady-state series simulations

Quick start basics - Your first simulation session .

Often, "parameter sweeps" are applied to determine the relation between engine performance and a specific parameter, for example fuel flow from idle to max power. Therefore, steady state operating point calculations are required over a wide operating range. GSP's "steady state series calculations" offers a flexible way to apply input parameter sweeps using input tables of e.g. controls.

Again, let's take the fuel flow to perform a parameter sweep.

1. Click in the Project tree panel on the "TJET_Config" node to load the original configuration. Now add a new case model (name it "Off-design sweep case") and set the case type to `St.St. Series` (confirm to save changes).
2. Double-click the manual fuel control icon of the model panel, or alternately right click on the icon and select `Edit` from the pop-up menu. The component data window for the manual fuel control component is now open.
3. Click the `St. St. OD Series` tab sheet. Check if a check mark is placed in the `Active` option to enable series.
   Here the Point-function table for fuel flow is visible. At the top row at Point `0` the design fuel flow should be specified (0.38 kg/s)
4. Click button A in  to add a line and enter `10` as point value.
5. Enter `0.1` in the fuel flow column `Wf[kg/s]` in the last row (click on it to edit) and click `OK`.
   Now a fuel flow range has been specified from maximum thrust down to near-idle thrust.
6. Click button `x` in 

   to clear the steady state output table (click `OK` in the optional Confirm window).
   It is always recommended to clear the output table before restarting a simulation (the reason for this will become clear later in graphical output).

7. Run the simulation to perform the parameter sweep calculation (click `OK` in the `set start time` window to confirm the starting point 0).
   The steady state output table is filled with the subsequent operating point data at each step (output and calculation interval can be set in Transient/Series options).
8. Optionally click `Graph` to obtain a graphical presentation of the results, see graphical output.
9. Repeat the process for similar parameter sweeps with other input parameters.
   For example, click `Amb.Cond.` , select `St. St. OD Series`, specify a Mach number parameter sweep and perform similar actions as above.

Hint: you may want to Reinitialize the model prior to a steady-state series simulation in order to bring the model state back to Design point before new input is specified.

5.1.6 Specifying transient simulation input

Quick start basics - Your first simulation session .

For transient simulations, input parameters need to be specified as a function of time. The engine response to that input change is then calculated. In GSP the input-time functions are specified using time-input tables in the input specification windows.
Again, let's take the fuel flow to determine the response in time. Assume the engine response on the following fuel flow vs. time function is desired:

- a fuel flow change from 0.38 (max. thrust) down to 0.1 in 1 second (deceleration),
- a 5 seconds stabilisation at this fuel flow,
- a fuel flow increase to 0.38 again (an acceleration) in 1 second
- finally followed by 5 seconds stabilisation again.

.

To obtain this input, do the following:

1. Click in the project tree panel on the `ReferenceModel` node (configuration model) to load the original configuration. Now add a new case model (name it "Transient case") and set the case type to `Transient` (confirm to save changes).
2. Enter the Fuel Control input dialog window (double-click the icon). Click the `Transient` tab sheet.
   Here the time-function table for fuel flow is visible. At the top row at time 0.000 the design fuel flow is already specified.
3. Click `A` to add a line and enter `1` as the time value.
   When analyzing transient performance, an engine response calculation is usually started after a short period of running steady state, in this case 1 second at the design steady state operating point.
4. Enter `0.38` in the fuel flow column `Wf[kg/s]` in the last row (click on it to edit).
5. Continue steps 3 and 4 to obtain the table shown in the figure below. Edit the time-fuel flow table using the navigation controls to go to the first line, previous line, next line or last line or to add, insert or delete a line in the table, or to delete the entire table. Note that fuel flow changes in infinite short times are invalid: a gas turbine engine control system would never be able to realize it and GSP will likely not be able to simulate it. So make sure you enter time values with reasonable (not too small) positive increments between the rows.

Time [s]	Wf [kg/s]
0.000	0.380
1.000	0.380
2.000	0.100
7.000	0.100
8.000	0.380
13.000	0.380

6. Optionally click `Graph` to view the transient input curve.
7. Check if transient input activated is checked and click `OK` (put a check mark in `Active` checkbox to activate the series).

Hint: you may want to Reinitialize the model prior to a steady-state series simulation in order to bring the model state back to Design point before new input is specified.

5.1.7 Running transient simulations

Quick start basics - Your first simulation session .

After specifying the fuel flow versus time function, run the transient simulation:

1. Run the simulation.
   (click `OK` in the `set start time` window and `Yes` in the `stabilize` window).
   Now, the transient output table pops up. During the calculation, subsequent transient operating points are added to the table at predefined time intervals (Transient/Series options).

Time [s]	Comment	W1 [kg/s]	TT2 [K]	PT2 [bar]	WC2 [kg/s]	TT3 [K]	PT3 [bar]	PR_2 [-]	N1 [rpm]	N%1 [%]
11.400		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
11.600		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
11.800		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
12.000		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
12.200		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
12.400		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
12.600		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
12.800		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88
13.000		19.8805	288.15	1.01325	19.8806	542.08	7.00471	6.9131	16521	99.88

66 records

Steady-state table | Steady-state graph | Transient table | Transient graph

s 3 Broyden 3 Jac.Inv. Time:13.050 sec

4. Optionally click `Graph` to obtain a graphical presentation of the results, see graphical output.
5. Repeat the process for similar transient calculations with other input parameters.
   For example, click `Amb.Cond.` in the model window, select `Transient`, specify a Mach number vs. time function to study engine response on changing Mach number and perform similar actions as above.

Note: The transient results calculated here with the TJET model do not represent typical transient gas turbine performance, due to the use of a Manual Fuel Flow Control component. For actual transient performance analysis a GSP generic or dedicated fuel control system component needs to be used instead.

5.1.8 Output formats

Quick start basics - Your first simulation session .

In the previous topics, simulation results were presented in tables. Often, tabular data does not provide a convenient representation of the results. GSP provides efficient methods of graphically visualising relations amongst parameters from the output tables. Furthermore, GSP can output individual operating point data to a formatted (ASCII or rich text format) text report.

5.1.9 Tabular output

Quick start basics - Your first simulation session .

The output tables provide a flexible format to share results. This can be done in two ways:

- For future use in GSP, click `Save` in the table window to save output in FireBird (SQL) database table format.
- For immediate use in other applications, right-click on the table and select `Copy table to clipboard` to copy the table to the clipboard, after which the data can be pasted in other applications such as spreadsheets and word processors.

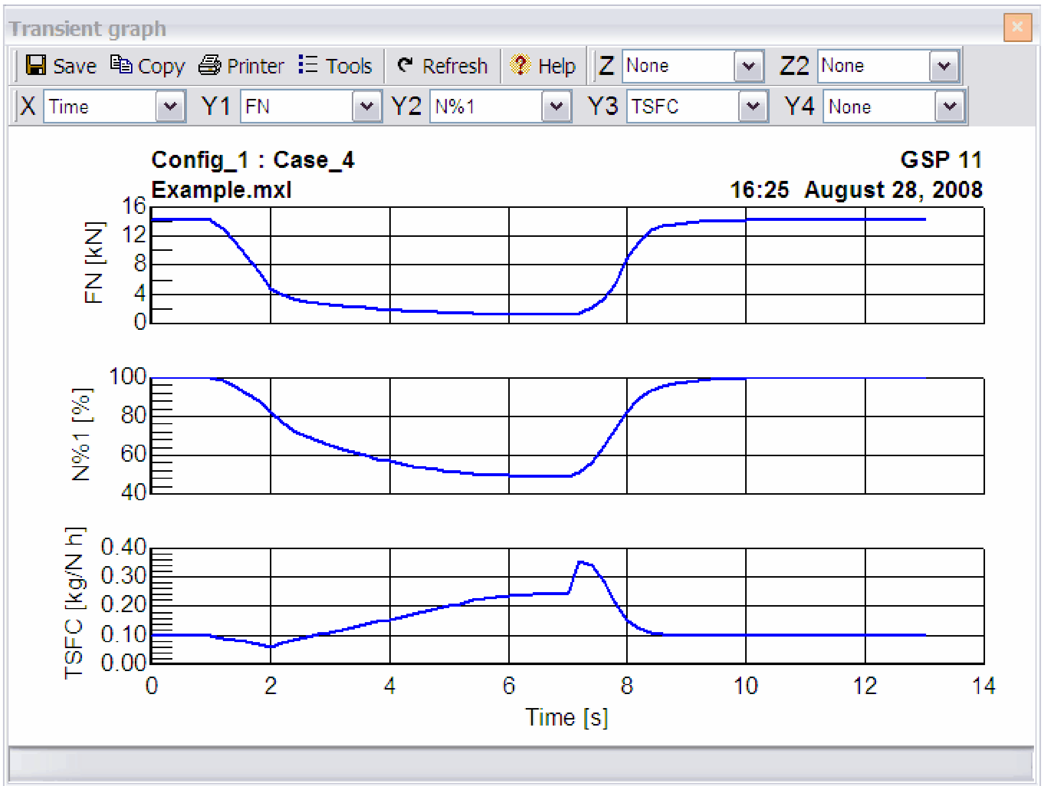
5.1.10 Graphical output

Quick start basics - Your first simulation session .

With GSP it is easy to graphically visualize relations amongst parameters from the output tables. This can be done for a maximum of 4 individual parameters as a function of 1 common parameter.

Let's graphically analyze the thrust, corrected turbine mass flow and TSFC as a function of time for the transient condition specified in Specifying transient simulation input.

1. If the transient table is not visible, select `Results|Transient|Table` of the project window. If the transient data is cleared, perform the actions as specified in Running transient simulations.
2. Click `Graph` in the table window.
   The transient table graph appears.
3. Select `FN` for the first y-axis *Y1* by clicking  and selecting it from the pull-down list.
4. Repeat step 3 for `N%1` as 2nd y-axis and `TSFC` as 3rd y-axis.
   If the transient graph panel is docked and too small to view the graphical results, undock (imply drag the panel by the tab) the panel and resize it to a larger size. The transient table graph shown below (shown undocked) displays the graphical output of the transient simulation.



5. Optionally select `Tools|Options` from the graph window menu to set general graph parameters including scaling, labeling or line styles.
6. Optionally select `File|Save graph as BMP file` or `File|Save graph as Meta file` from the graph window menu to save the graph to disk. Alternately select `File|Copy Bitmap to clipboard` or `File|Copy Meta file to clipboard` from the graph window menu to copy the graph to the clipboard for immediate use in other applications. For enhanced use in for instance word processors, you are recommended to save graphs as meta file.
7. Repeat the process for similar graphs with other output parameters.
   Especially with transient data it is interesting to select other parameters besides time as x-axis.

Note: It is advised to make sure the table exclusively contains the data of interest, since non-relevent data may spoil the graphs. Use X *in the table window to clear the table before the relevant simulation is started.*

5.1.11 Component operating line

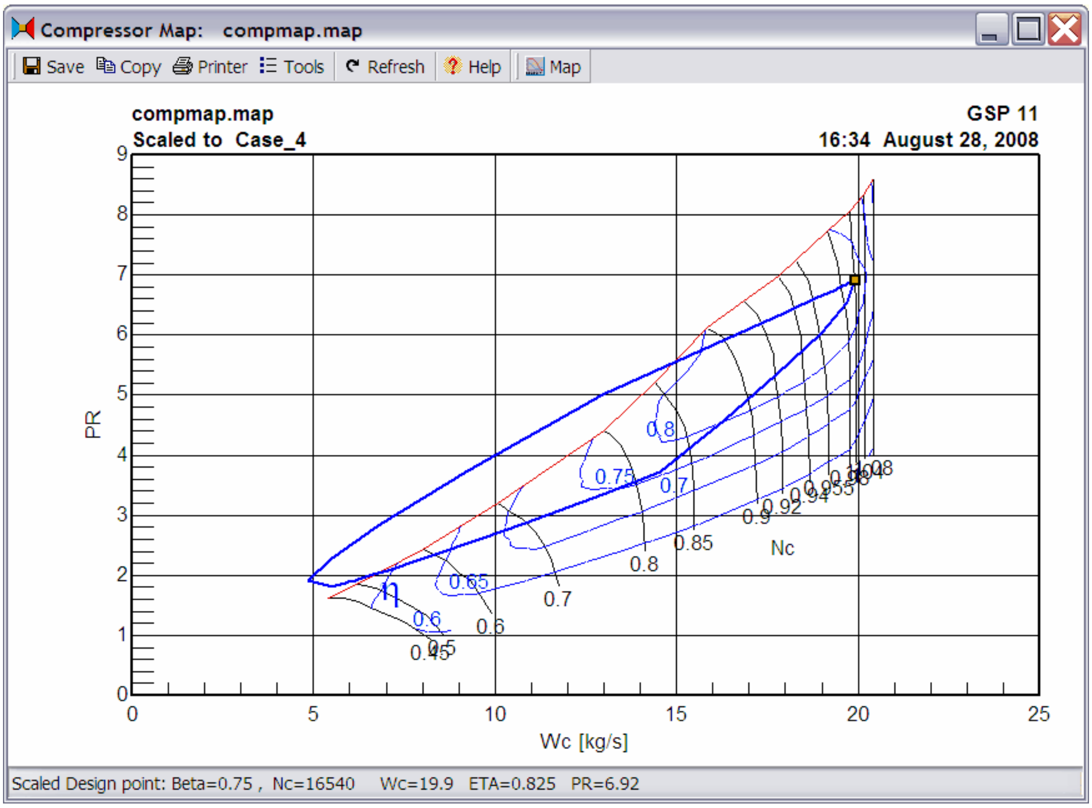
Quick start basics - Your first simulation session .

With GSP you can view the steady state and transient operating lines in the components using maps, like the compressor and turbine.

Let's view the operating line for the transient condition specified in Specifying transient simulation input:

1. Enter the Compressor component data window (double-click the compressor icon).

2. Select the `Map` tab sheet.
   This tab sheet contains the filename and design settings for the off-design component map.
3. Click the show graph button of the map tool bar to show the compressor map.
4. If unchecked, check `Map|Scale to Model des. pnt.` by clicking on it.
   Now the compressor map is scaled to the model design point. In practice, component maps for specific engines are hard to obtain. For this reason generic maps are used, obtained from similar components having a different scaling. To use the component map it is scaled in the design point for which all values used in the map are known/given.
5. Click `Map|Draw Transient` to display the transient operating line.
   The graph shown below is the result. If an error message is displayed, check if `Map operating curve par.s` is checked in the compressor Output tab sheet. If not, check it and redo the actions from Running transient simulations.



As can be seen from the graph, the used model is a simplified model, since the operating line is such that the compressor is stalling. In real practice this would be prevented by several bleed measures.

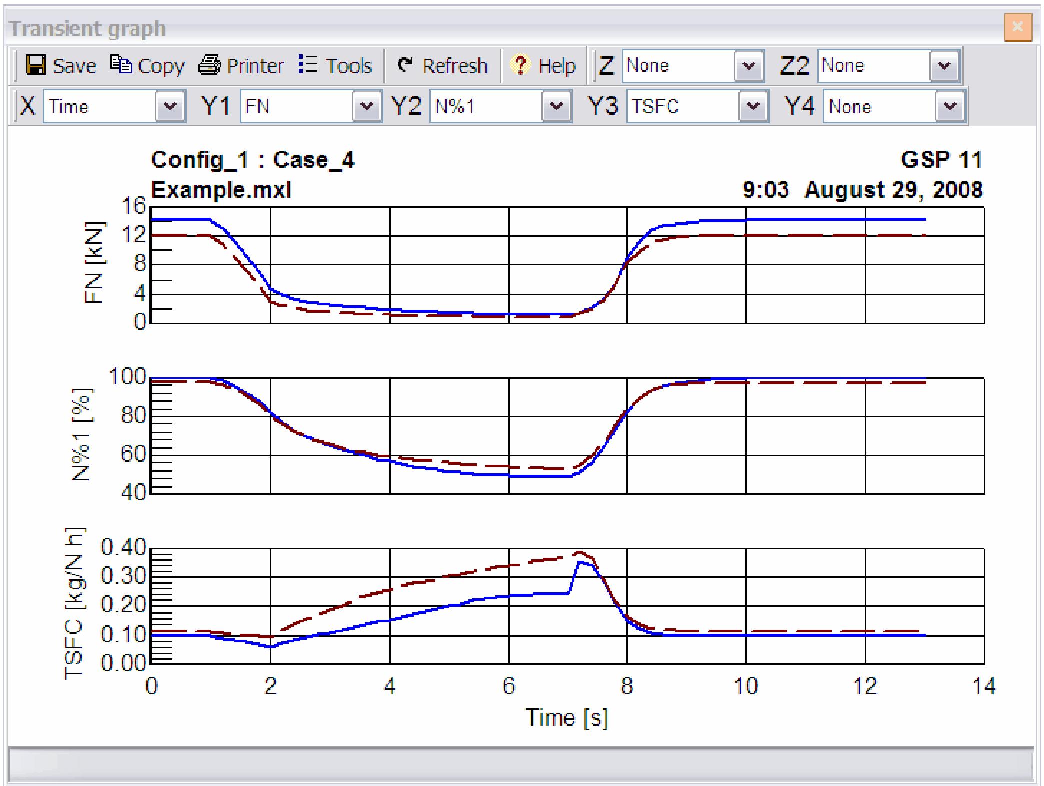
5.1.12 Multiple output curves

Quick start basics - Your first simulation session .

An important feature of the graphical output visualization is the option to present multiple steady state or transient series for a specific parameter in one graph using breaks.

Let's take a look at the effect of Mach number on the transient performance as specified in Specifying transient simulation input.

1. If the transient table is not visible, select `Results|Transient|Table` in the model window. If the transient data is cleared, perform the actions as specified in Running transient simulations.
2. Click the 'append break button' of the table window toolbar to append a break line. The graph window will automatically interpret the break lines as curve separators.
3. Click `Amb.Cond.` above the project tree.
4. Click the `Off-design conditions` tab sheet.
5. Change the `Air Speed Mach` value to `0.4` and click `OK`.
6. Run the model to perform the transient calculation (set time to 0, click `OK` in the `set start time` window and `Yes` in the `stabilize` window).
7. Click `Graph` in the transient table window.
   The graph is displayed, with 2 curves for Mach 0.0 (blue) and 0.4 (black dashed).



8. Optionally change axis variables for similar graphs with other output parameters.
9. Repeat the process for similar graphs with other input parameters.
   For example, study the effects of height (zp) on performance.

Note: It is required to "reset" the operating point of the model between the steady state series or transient simulations. Make sure extra data lines resulting from any steady state or design point calculations are deleted to prevent spoiling of the graph.

5.1.13 End of first simulation session

Quick start basics - Your first simulation session .

If you have performed all specified tasks, then:

Congratulations !
You have completed your first GSP gas turbine performance simulation session.

The other sample models provided with GSP show you how to build models for more complex gas turbine engine configurations, including control systems, variable geometry nozzles, bleed valves, power turbines and recuperators.

If you only want to use GSP to run existing models developed by NLR or others, you do not need continue this tutorial. If you want to

- analyze effects of changing engine characteristics or
- learn how to build your own GSP gas turbine models,

please continue this tutorial with the next section.

5.2 Adapting engine characteristics

Quick start basics - Adapting a model .

After having analysed steady state and transient performance of an existing model, adapting the model to another type of engine with the same configuration might be interesting. In that case, component model parameters need to be modified, resulting in different engine performance characteristics.

A model can easily be modified by double-clicking on the component icons and editing the various component data. You may select different modeling options, change general component data, modify the design point by changing design point values (indicated by a **navy blue color** in Windows standard colors, not using the XP theme since the XP theme overrides custom colors), or change off-design behaviour by using different component maps. A model may also be extended with additional components, like extra booster-compressors (to be inserted between fan and compressor) in case of a turbofan configuration, or with controls for bleed flows. This more advanced use of GSP is demonstrated in "Deriving a new model".

5.2.1 Effects of changes in engine characteristics

Quick start basics - Adapting a model .

It is often required to analyze the effects of small changes in the engine characteristics relative to a *baseline* engine.

For example, to analyze the effect of a lower compressor isentropic efficiency on the TJET model transient performance. Effectively another engine design is obtained. To model this modified engine do the following:

1. Click in the Project tree panel on the "TJET_Config" node to load the original configuration. Now insert a new configuration and name it "Alternate engine"
2. Double-click the compressor component in the model panel and change the design efficiency from `0.825` to `0.75.`
3. Altering the design should be noted on the model notes panel to state any deviations from the base/reference model so that other people understand the model structure. Enter a not similar to: "This model has a lowered compressor efficiency due to using a more conservative compressor design. The design efficiency has changed from 0.825 to 0.75."
4. Create a child transient case and specify a transient fuel flow input function in the model as specified in "Running transient simulations".
5. Click the 'append break button' of the table window toolbar to append a break line. The graph window will automatically interpret the break lines as curve separators.
6. Run the simulation.

The transient table now contains two transient response calculations, separated by a break row (first transient dataset was created in Running transient simulations).
The graph will now display two datasets, so that the performance of the alternate engine can be graphically compared to the initial design.

Note: The transient results calculated here do not represent typical transient gas turbine performance due to the use of a Manual Fuel Flow Control component. For actual transient performance analysis a GSP generic or dedicated fuel control system component needs to be used instead.

5.2.2 End of second simulation session

Quick start basics - Adapting a model .

If you have performed all specified tasks, then:

Congratulations !
You have completed your second GSP gas turbine performance simulation session and analyzed the effect of changing engine characteristics.

If you only want to use GSP to run existing models developed by NLR or others or want to analyse changing engine characteristics, you do not need continue this tutorial. If you want to

- learn how to build your own GSP gas turbine models,

please continue this tutorial with the next section.

5.3 Building a GSP engine model

Quick start basics - Building a GSP model .

Now that you have completed your first GSP simulation sessions and you are a more experienced GSP user, you can start building your own GSP models. It usually is most practical to derive new models from existing or sample models by opening an existing model and adding, deleting or reconfiguring components as necessary.

In this tutorial we will derive an afterburning turbofan engine from the sample TJET model used in "Your first simulation session" Assume that the following additional data must apply to the new engine model at ISA sea level standard (SLS) static conditions:

Thrust SLS	>50	[kN]
Design mass flow	90	[kg/s]
Bypass ratio	0.64	[-]
Fan pressure ratio	3.2	[-]
Ram recovery	0.9	[-]
Fan efficiency	0.8	[-]
N1	10,000	[rpm]
N2	13,000	[rpm]

5.3.1 Adapting model configuration

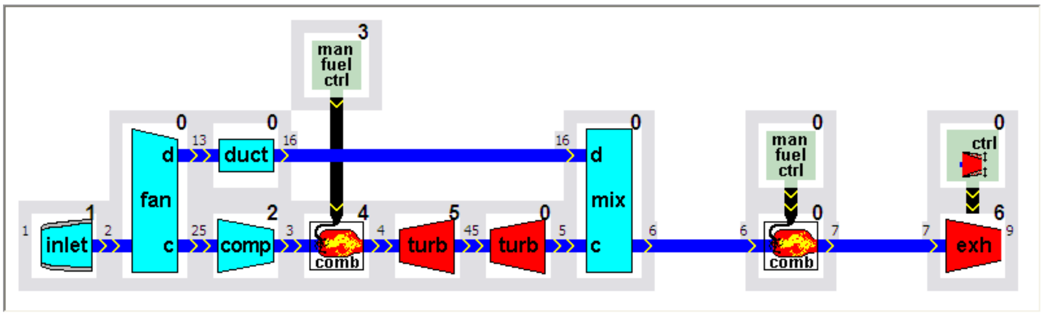
Quick start basics - Building a GSP model .

After you have started GSP and you have opened the TJET.mdl model (see "Opening a model"), or alternatively proceed from the project created in the previous two examples ("Your first simulation session" or "Adapting engine characteristics").

To obtain the afterburning turbofan configuration, do the following:

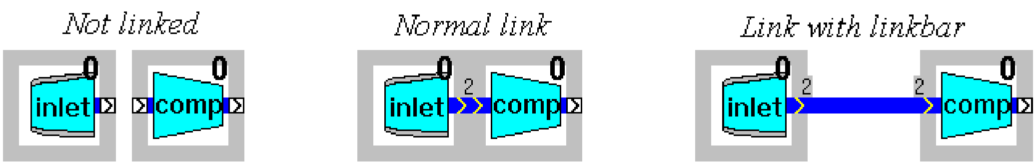
1. A new configuration needs to be created under the "ReferenceModel" to allow adding of components and setting of reference/design data. Name this configuration "ABFAN".

We are about to change this configurations engine configuration by adding the additional components to the model panel (drag-and-drop from the component libraries), and arrange them according to the desired displayed (below) gas turbine configuration. Look for appropriate components in the Gas Path Component Library (Duct, Turbine, Several component Link bars), Multi in/out Component Library (Fan, Mixer), Standard Controls Component Library (Manual afterburner fuel control, Manual Variable Exhaust Nozzle Control).



2. Select all components by clicking with the mouse inside the model panel area, keeping the mouse button pressed and drawing a rectangle around all components.
3. Move the components by clicking with the mouse on any component, keeping the mouse button pressed and moving the components to the right. Unselect all components by clicking somewhere in the model panel area, but not on one of the components.
4. Select the inlet component and move it to the left to make room to insert the fan component.
5. Select the fan component with the mouse from the Multi in/out Component Library, keep the mouse button pressed while dragging the component to the model panel and release the mouse button to drop the component on the form in between the inlet and the compressor.

6. Link the fan component to the inlet and enter 2 as the station number (if not done automatically).
   Components are linked by placing them adjacent to each other or using the link bar component. The requested station number is the designator of the output of the gasturbine component. If components are linked, the connection indicator turns dark blue and the station number appears as displayed.



7. Link the fan component to the compressor (move inlet and fan as stated above), unselect the components and set the station number to `25`.
   Be sure that no components are selected, otherwise no stations can be selected.
8. Select the duct component from the standard library, connect it to the fan bypass outlet and set the station number to `27`.
9. Select the fuel control component on the model panel and move it up to above the duct. If there is no room, enlarge the window, select all components and move them down.
10. Select the link bar component from the Gas Path Component Library and place it anywhere on the model window area, but not on a component.
11. Connect the fuel control component to the combustor component with the link bar. The length of the link bar can be changed by dragging one of the ends like the standard Windows enlarging operation. If successful, the link bar turns black. No station number is required.
12. Select, move and drop the remaining components and input and change the station numbers to create the model displayed above, with the exception of the exhaust control component.
    If the model panel is too small, simply enlarge it by dragging the window borders.
13. Select the exhaust component on the model window and enter the component data sheet by double-clicking the component or right-clicking it and selecting `Edit`.
14. Select the `General` tab sheet and check the `Variable area nozzle` radio button in the `Model Options`.
    The text `Use Nozzle control component !` appears.
15. Click the `OK` button to close the exhaust component data sheet.
    A connector is now visible at the top of the exhaust component icon.
16. Select the manual variable exhaust nozzle control component (VEN) from the control library and connect it to the exhaust component.

If the new model has been completely build, the component data must be entered.

5.3.2 Entering component data

Quick start basics - Building a GSP model .

After arranging the components according to the desired configuration, the individual component characteristics must be specified by entering data in the component model panels for the following components:

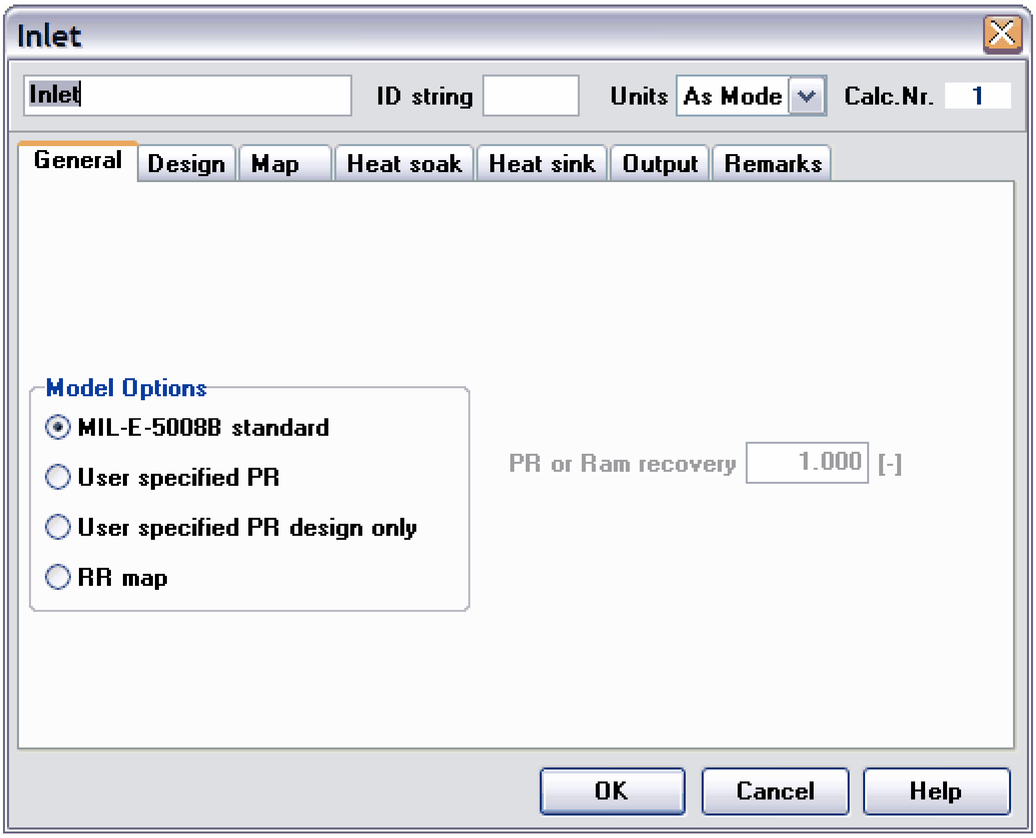
- inlet,
- fan,
- compressor,

- combustor,
- HP (high-pressure) turbine,
- LP (low-pressure) turbine,
- mixer,
- afterburner.

5.3.3 Entering inlet data

Quick start basics - Building a GSP model .

1. Select the inlet component and double-click on it, or right-click on it and select `Edit`.
   The inlet component data window appears

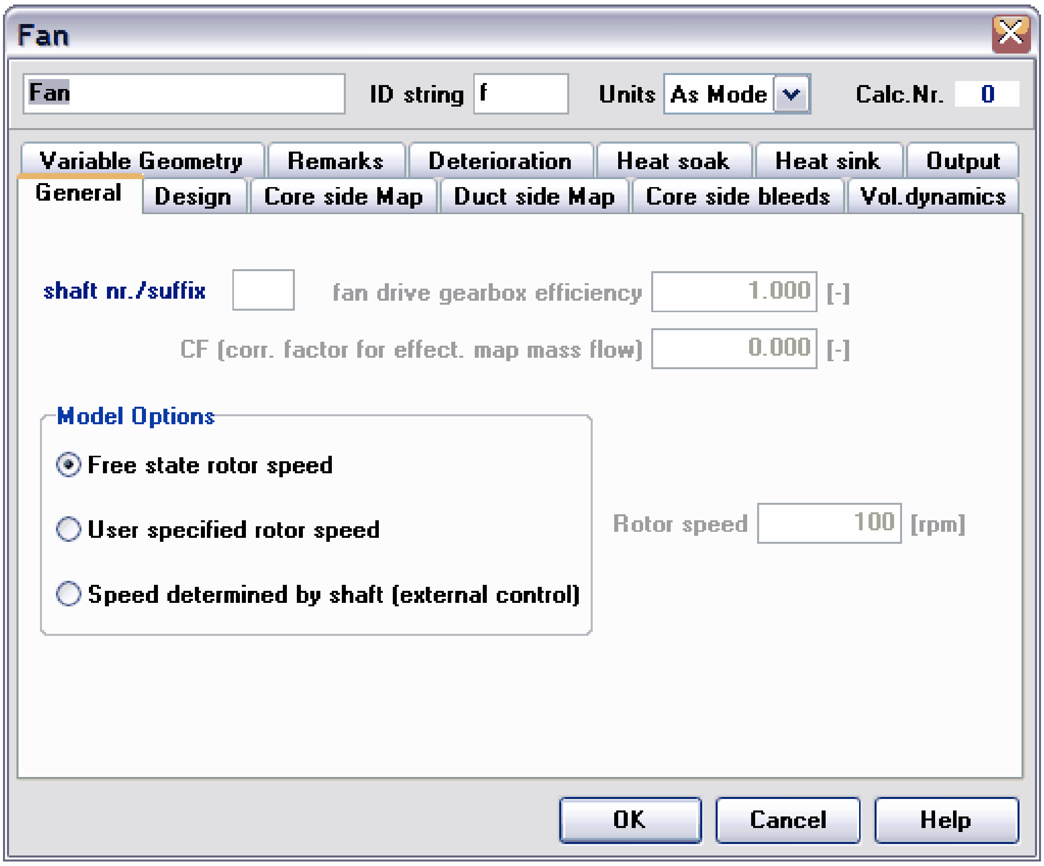


2. Select the `Design` tab sheet.
   This tab sheet contains all variables required to configure the compressor design for the engine model.
3. Set the `Design mass flow` to `90` kg/s.
   The afterburning bypass engine designed here requires a much higher mass flow than the original TJET engine does, since the core engine will not be changed.
4. Set the `Pressure ratio` to `0.9`.
5. Click `OK` to close the inlet component data form

5.3.4 Entering fan data

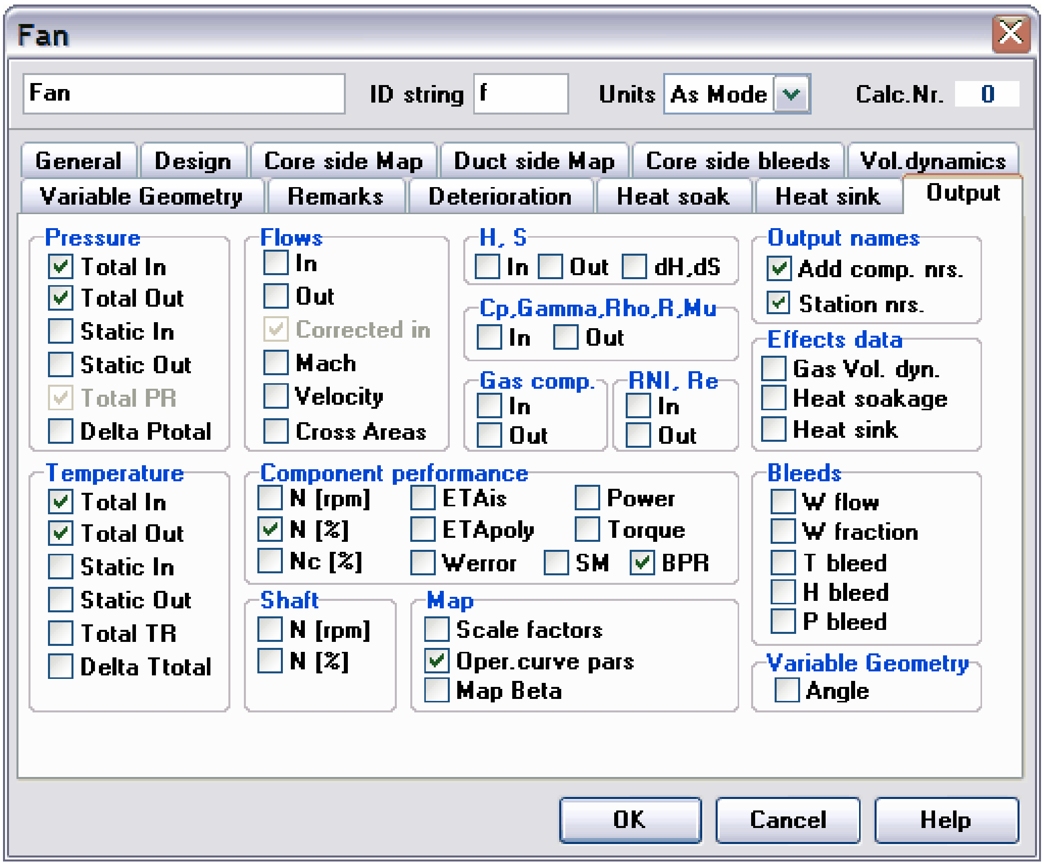
Quick start basics - Building a GSP model .

1. Select the fan component and double-click on it, or right-click on it and select `Edit`.
   The fan component data form appears



2. Select the `General` tab sheet and check that the `shaft nr./suffix` is set to `1`.
   With this the fan is linked to the low pressure turbine, if the same shaft number is specified.
3. Select the `Design` tab sheet.
   In this tab sheet all major design variables for the fan are specified.
4. Set the `design bypass ratio` to `0.64`.
5. Set the `design rotational speed` to `10000` at `100` %.
6. Set both the `core side design pressure ratio` and `duct side design pressure ratio` to `3.2`.
7. Set both the `core side design efficiency` and `duct side design efficiency` to `0.8`.
8. Select the `Core side Map` tab sheet.
   Here the map file for off-design calculations and the fan design point are specified.
9. Click the `Browse map files` button, select the `ABFANfanc.map` file from the GSP standard directory and click the `Open` button. Select the storage switch to `Local`.
10. Set the map `design Beta value` to `0.6` and check if the `map design rotor speed` is set to `1`.
    Now the fan core design point is set. The Beta parameter is a dimensionless map parameter explained in beta parameter.
11. Optionally click the `Show Graph` button to view the component map and design point.
12. Select the `Duct side Map` tab sheet and repeat 9 to 11 with the `ABFANfand.map` file.

13. Select the `Output` tab sheet
    In this tab sheet the requested output parameters for the used component are specified.

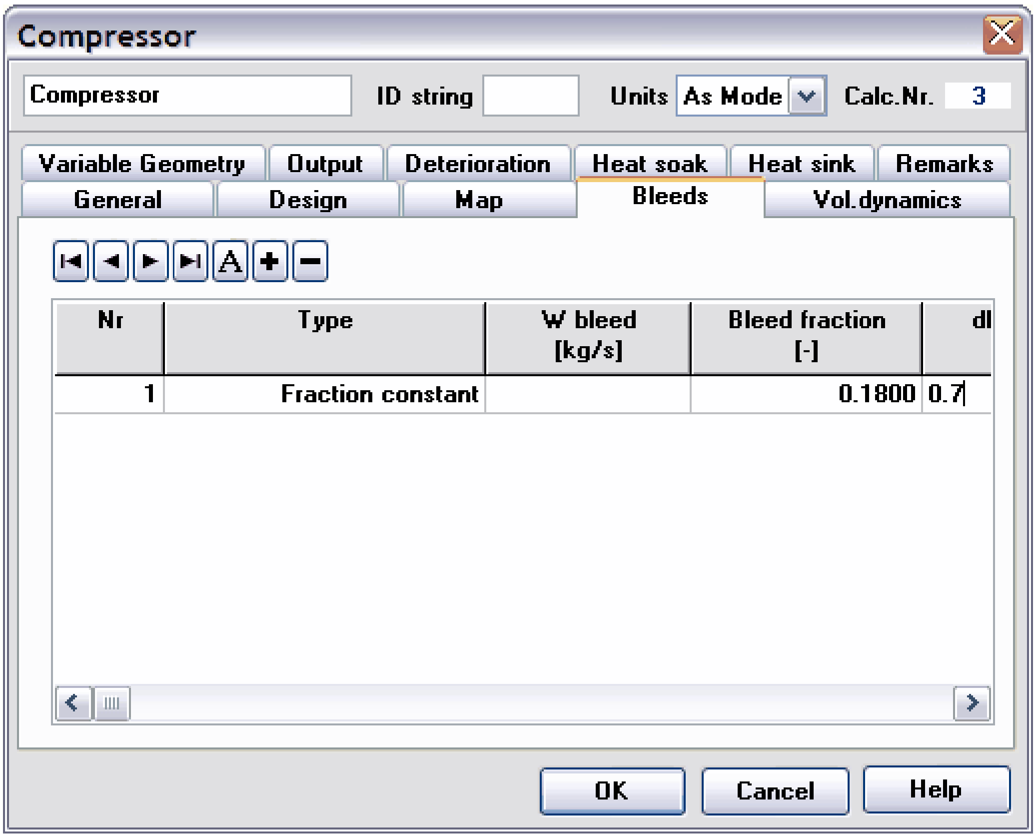


14. Check `Pressure Total Out`, `Temperature Total Out`, `Flows Corrected in`, `Turbo N [%]` and `Oper. curve pars`.
    In the output table which will be generated after simulation, the total pressure and total temperature at the outlet of the fan core and duct are reported, as is the corrected mass flow at fan entry. By checking the `Oper. curve pars` several variables are automatically selected, enabling the viewing of the operating line in the off-design map.
15. Click `OK` to close the fan component data form.

5.3.5 Entering compressor data

Quick start basics - Building a GSP model .

1. Open the compressor data form by selecting the compressor and double-clicking on it, or right-clicking on it and selecting `Edit`.
2. Select the `General` tab sheet and set the `shaft nr./suffix` to 2.
   Since the fan is already added, which should be connected to a turbine, and the shaft number for the fan is set to 1, this specifies that there are two shafts.
3. Select the `Design` tab sheet
   Here, design parameter values are specified with which the engine is designed.
4. Set the `design rotational speed` to `13000` rpm at `100%`
5. Select the `Bleeds` tab sheet.



This engine incorporates turbine blade cooling for which air is bled from the compressor.

6. Set the `Type` of bleed `Nr. 1` to `Fraction constant` with the pull-down menu. The amount of bleed flow is now a fraction of the compressor mass flow. Note that depending in the Type selected, the `W bleed` or `Bleed fraction` columns can or cannot be edited. With the `Externally controlled` option, only `dH fraction` can be set.
7. Set `Bleed Fraction` to `0.18`. The compressor mass flow fraction used for bleed is now specified.
8. Set `dH Fraction` to `0.70`. This specifies the point in the compressor from where the bleed flow is bled.
9. Select the `Map` tab sheet to embed the map file by setting the storage location to `Local` if not already done so.
10. Select the `Output` tab sheet and check the output variables desired.
11. Click `OK` to close the compressor component data form.

5.3.6 Entering combustor data

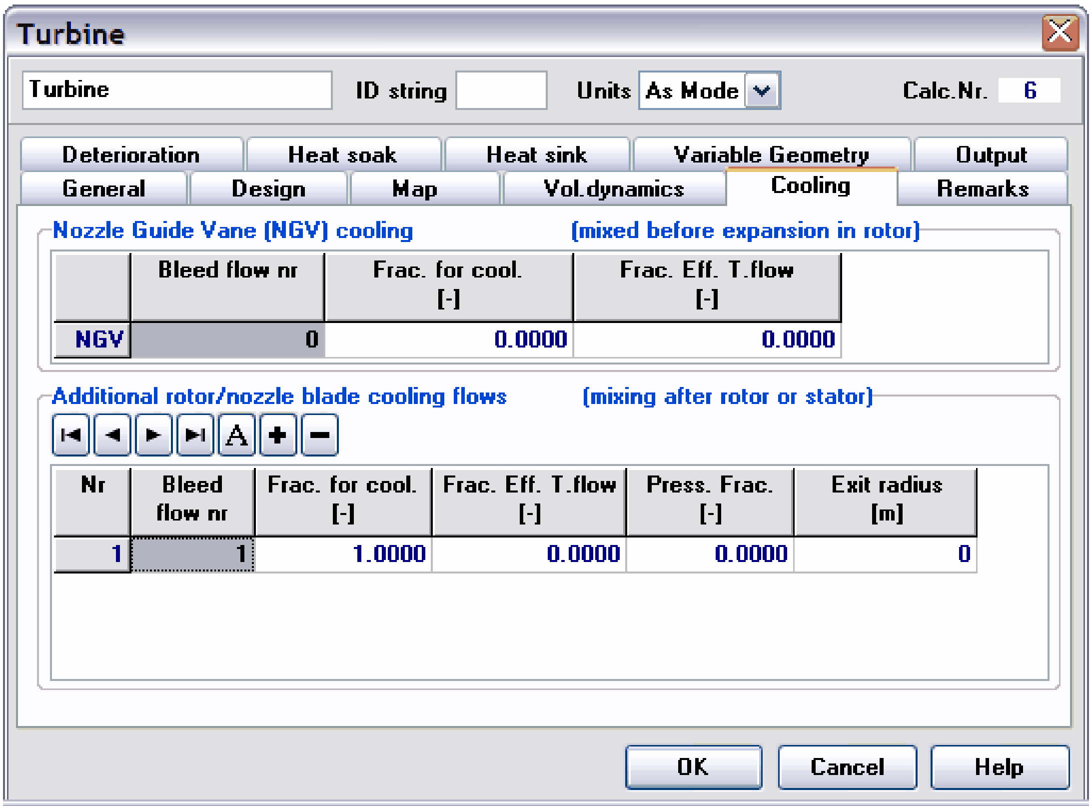
Quick start basics - Building a GSP model .

1. Open the general combustor data form and select the `Design` tab sheet.
2. Set the `design fuel flow` to `1.2` kg/s. Since the mass flow through the engine is increased, the fuel flow must increase.
3. Select the `Output` tab sheet and check the output variables desired.
4. Click `OK` to close the combustor data form.

5.3.7 Entering HP turbine data

Quick start basics - Building a GSP model .

1. Open the high pressure turbine data form.
2. Select the `General` tab sheet and set the `shaft nr./suffix` to `2`.
   Now the high pressure turbine is linked to the compressor.
3. Select the `Design` tab sheet and set the `design rotational speed` to `13000` rpm at `100%`, equal to the compressor rotational speed.
4. Select the `Cooling` tab sheet.



Here the amount of bled air used for cooling is specified.

5. Set the `Bleed flow nr` for cooling `Nr.1` to `1`.
6. Set the `Frac. for cool.` for cooling `Nr.1` to `1.00`.
   Now the total amount of air bled from the compressor is used for turbine blade cooling.
7. Select the `Map` tab sheet to embed the map file by setting the storage location to `Local`.
8. Select the `Output` tab sheet and check the output variables desired.
9. Click `OK` to close the high pressure turbine component data form.

5.3.8 Entering LP turbine data

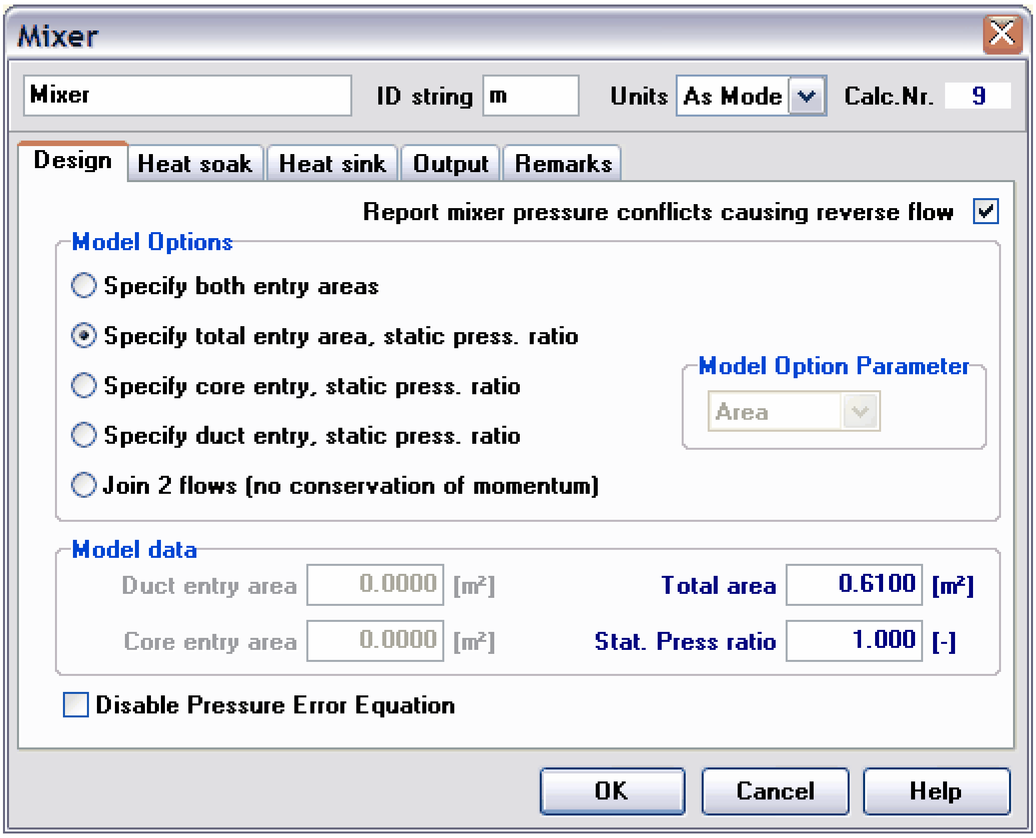
Quick start basics - Building a GSP model .

1. Open the low pressure turbine data form.
2. Select the `General` tab sheet and check if the `shaft nr./suffix` is set to `1`. Now the low pressure turbine is linked to the fan.
3. Select the `Design` tab sheet and set the `design rotational speed` to `10000` rpm at `100%`, equal to the fan rotational speed.
4. Select the `Map` tab sheet and browse for a map '`turbimap.map`' to embed the map file by setting the storage location to `Local`.
5. Select the `Output` tab sheet and check the output variables desired.
6. Change the NGV station label (on the `output` tab) from 41 to 47 to prevent station label duplication.
7. Click `OK` to close the high pressure turbine component data form.

5.3.9 Entering mixer data

Quick start basics - Building a GSP model .

1. Open the mixer data form and select the `General` tab sheet.



In this tab sheet the mixer entry area is specified, either separately for duct and core entry, or combined as total entry area with a specific static pressure ratio.

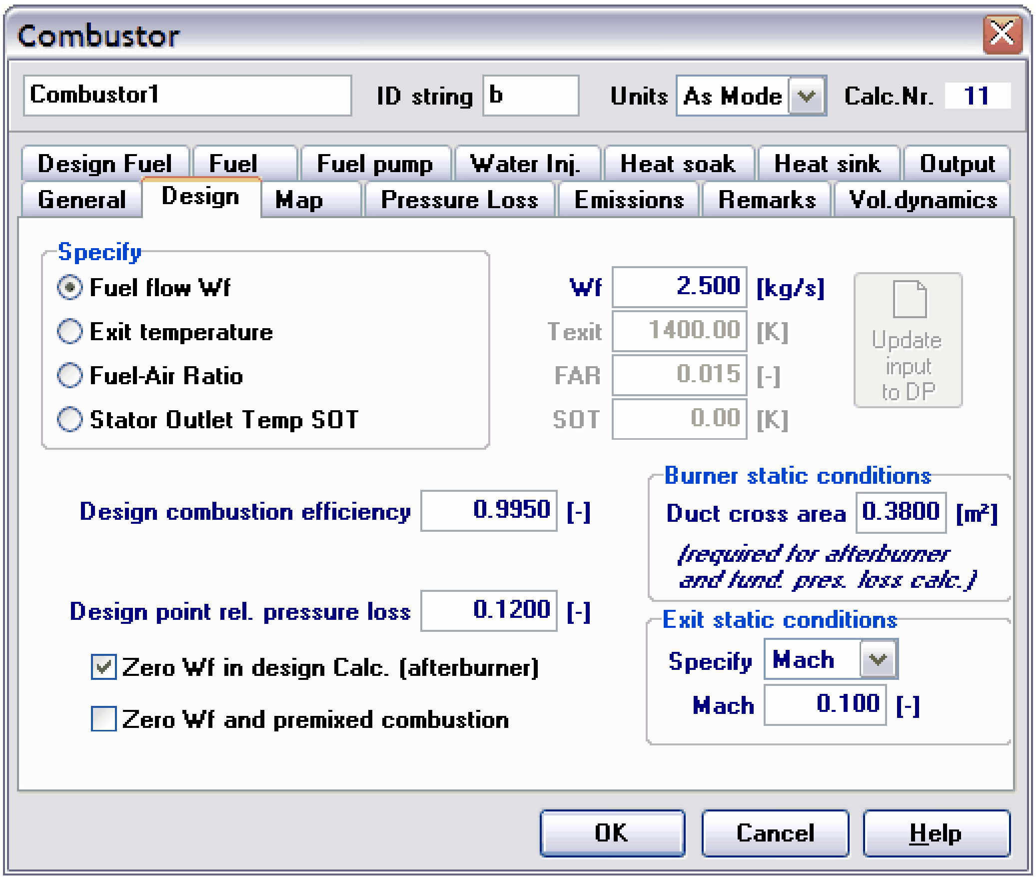
2. Select the `Specify total area, static press. ratio` radiobutton.
3. Set the `Total entry area` to `0.61`.

4. Set the `Stat. Press. ratio` to `1`.
   The static pressures for duct and core are matched.
5. Select the `Output` tab sheet and check the output variables desired.
6. Click `OK` to close the mixer component data form.

5.3.10 Entering afterburner data

Quick start basics - Building a GSP model .

1. Open the afterburner combustor data form.
2. Select the `General` tab sheet.
   Here it is selected whether the combustor is a normal combustor or an afterburner.
3. Select the `Use afterburner combustion efficiency maps` radiobutton.
   The `Burner duct cross area` variable appears.
4. Select the `Design` tab sheet.



5. Set the `design fuel flow` to `2.5` kg/s.
   The afterburner incorporates a significant higher fuel massflow than the normal combustor.
6. Set the `rel. tot. pressure loss at design point` to `0.12`.
7. Check the `Zero Wf in design Calc. (afterburner)` checkbox.
   When enabled, the engine is designed without afterburner, and the afterburner is only used during off-design simulation.
8. Select the `Map` tab sheet and browse for the following maps (from top to bottom):
   `ETAABfarrel.map`, `ETAABmrel.map`, and `ETAABprel.map` and set the storage option to `Local`.

9. Select the `Output` tab sheet and check the output variables desired.
10. Click `OK` to close the afterburner combustor component data form.

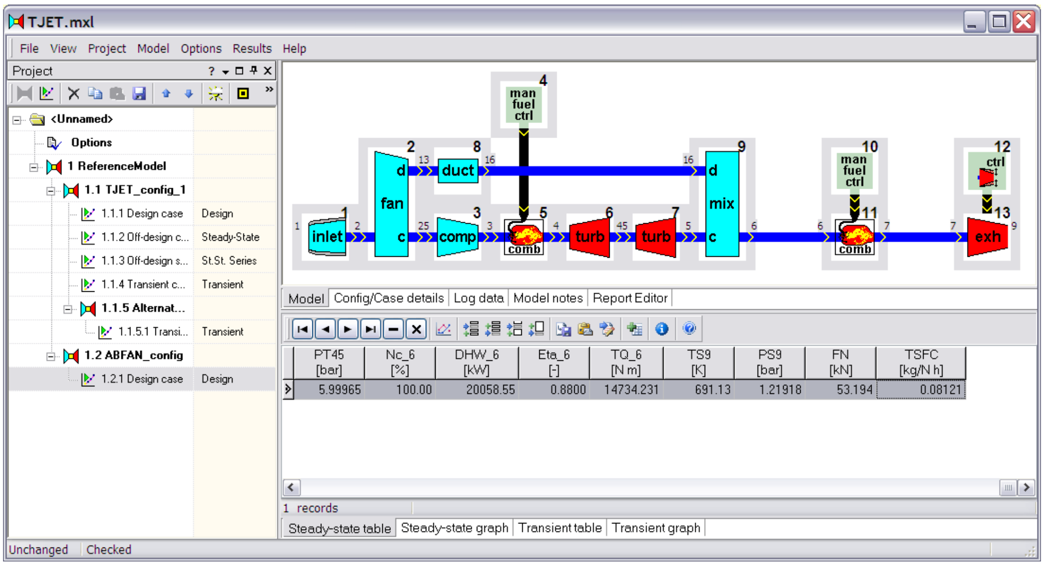
Now that the model has been completed, it is time to run simulations to test the validity of the model and analyze performance.

5.3.11 Running a new model

Quick start basics - Building a GSP model .

With the new built GSP model configuration simulations can now be run. Before a simulation is run the validity of the model configuration and component data is checked during the Design point calculation. If the model is invalid, the calculation is canceled and an error is reported (for possible errors see error messages). If the design point calculation succeeds, the model is valid and steady state, steady state series and transient simulations can be run as explained in "Your first simulation session".

The project should look similar to the figure below.



5.3.12 Quick start finish

Quick start basics - Building a GSP model .

If you have performed all specified tasks and you can simulate design, steady state and transient calculations, then:

Congratulations !
You have built your first model and performed simulation sessions with it.

The sample models provided with GSP show you how to build models for more complex gas turbine engine configurations, including control systems, variable geometry nozzles, bleed valves, power turbines and recuperators.

5.4 Performance Deck Generation

Quick start basics - Performance Deck Generation

In this section of the tutorial you will learn how to create a performance deck.

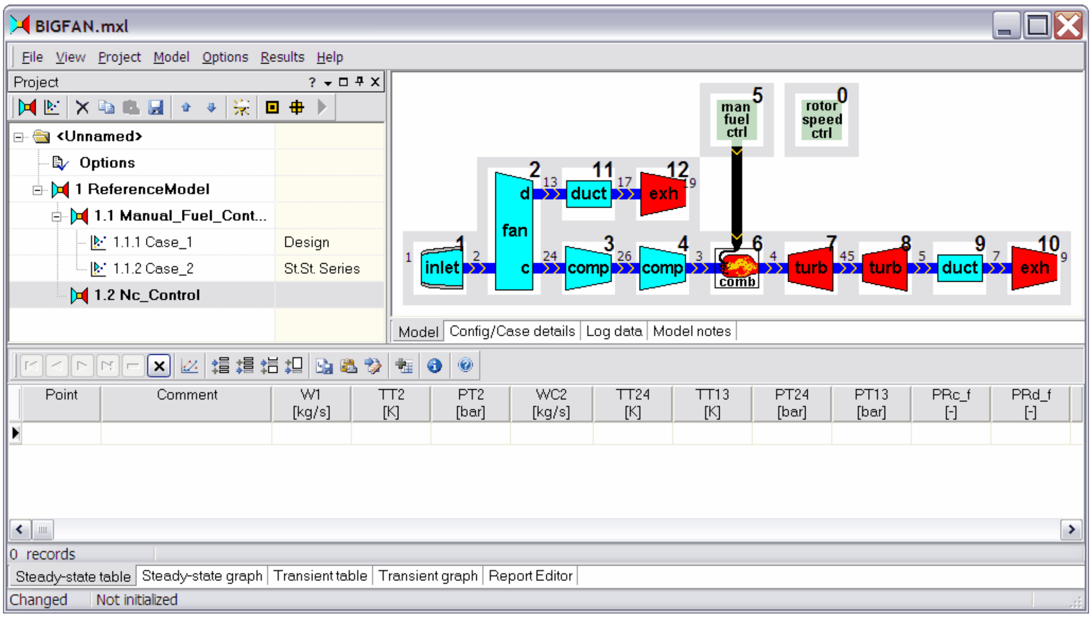
An existing model can be used to create the performance at various power settings of the engine throughout the flight envelope of the aircraft. This data can be used for off-line studies, e.g. to calculate the aircraft flight performance.

5.4.1 Adapting model configuration

Quick start basics - Performance Deck Generation

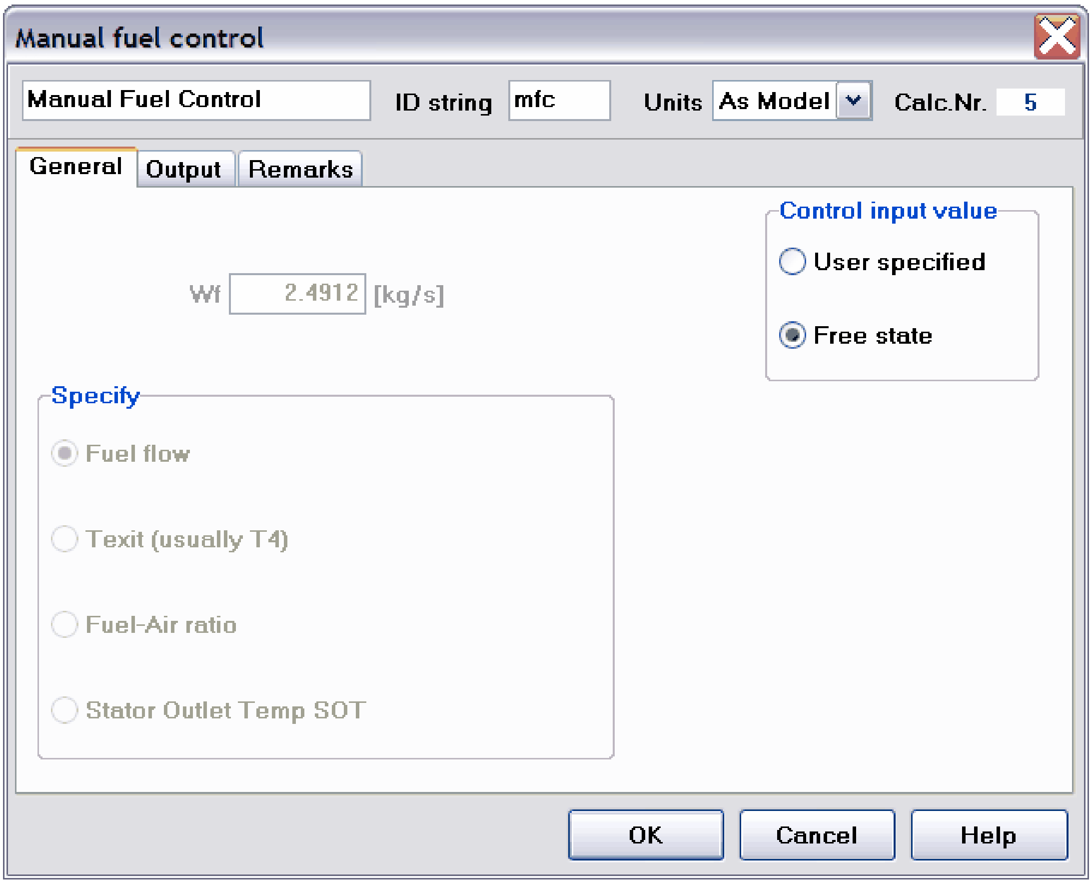
Open the `BIGFAN.mxl` project from the installed GSP projects folder and set the focus on the Base configuration model by clicking on the `Base Configuration` node (`ReferenceModel` in GSP 11) in the project tree panel.

This engine will be using a simple engine controller to demonstrate the integration of the controller in the engine performance flight envelope scheduler. We are therefore required to add a new `Configuration` to the `Base configuration model` node. Rename the configuration to "Nc_Control". Drag a rotor speed control component from the power control component library to the newly created configuration `Nc_Control`. The model window pane should resemble something like the figure beneath.

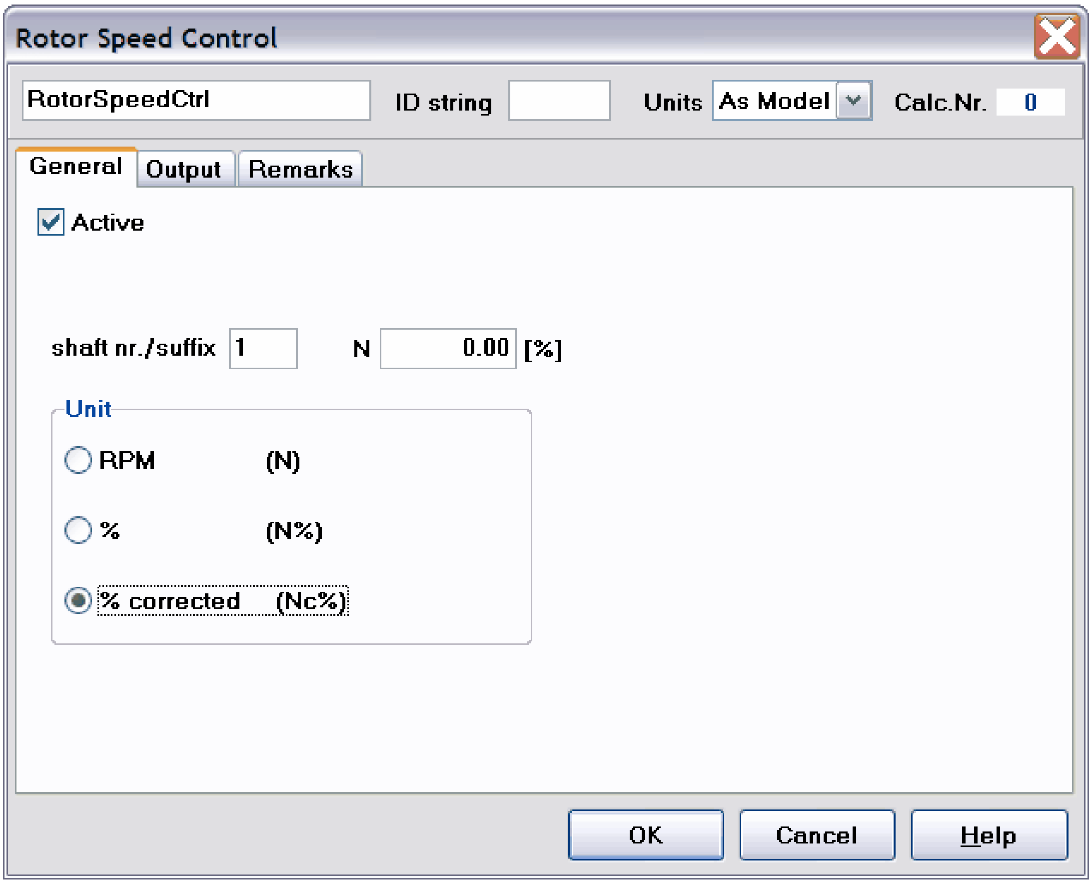


Currently the the model is not configured properly yet to use the rotor speed controller. A rotor speed controller is a component that adds an error equation to the equation system. The error is defined as the normalized difference between the actual and the demanded rotor speed setting. To solve the equation with this added error, a state variable has to be added. We therefore change the manual fuel controller from `user specified` to `free state`.

Open the data entry window of the manual fuel controller and set the control input value to free state:



Now configure the rotor speed controller by setting the `shaft nr./suffix` and the `Unit` to corrected rotor speed by selecting the `% corrected` option (Note that the shaft corrected speed corresponds to the speed corrected by the component that is the first in the primary airflow path of the specified shaft; in this case the fan, in case a different corrected rotor speed is required for the control it is best to replace the rotor speed control component with a generic equation schedule component):



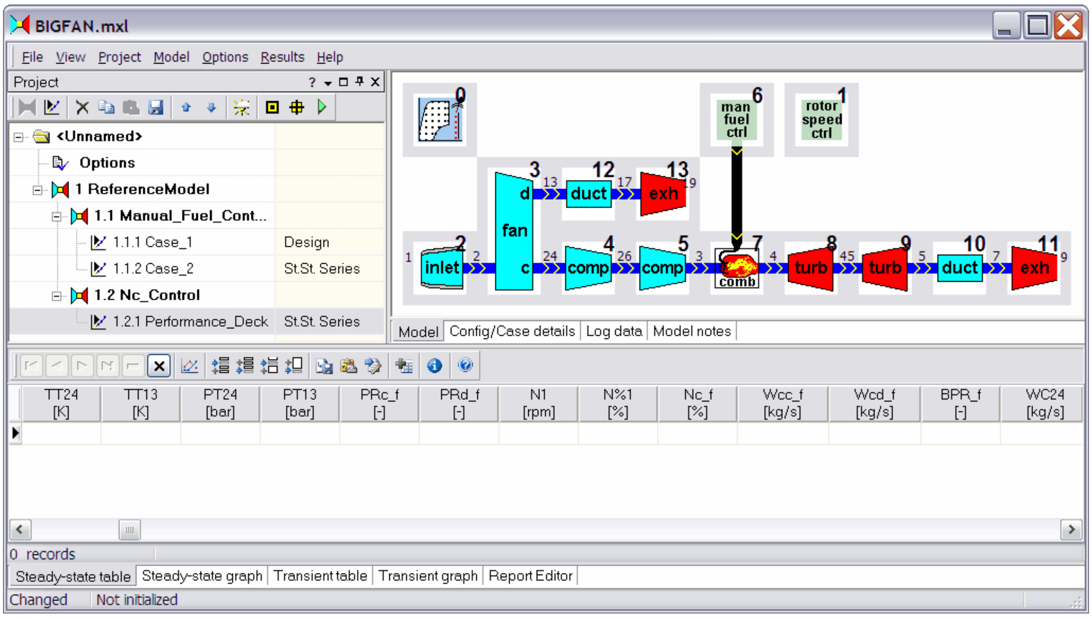
Note that this controller is a simple representation of an engine control system. An engine control is a sophisticated system which can be modeled more accurately by GSP. A more accurate control system can be created by exploring the known limits of the engine like maximum speed, temperature and pressure, and translate these to control schedules (usually the advanced power control component can be used for this since it is able to set a certain power setting as function of map scheduled parameters).

5.4.2 Flight Envelope Scheduler

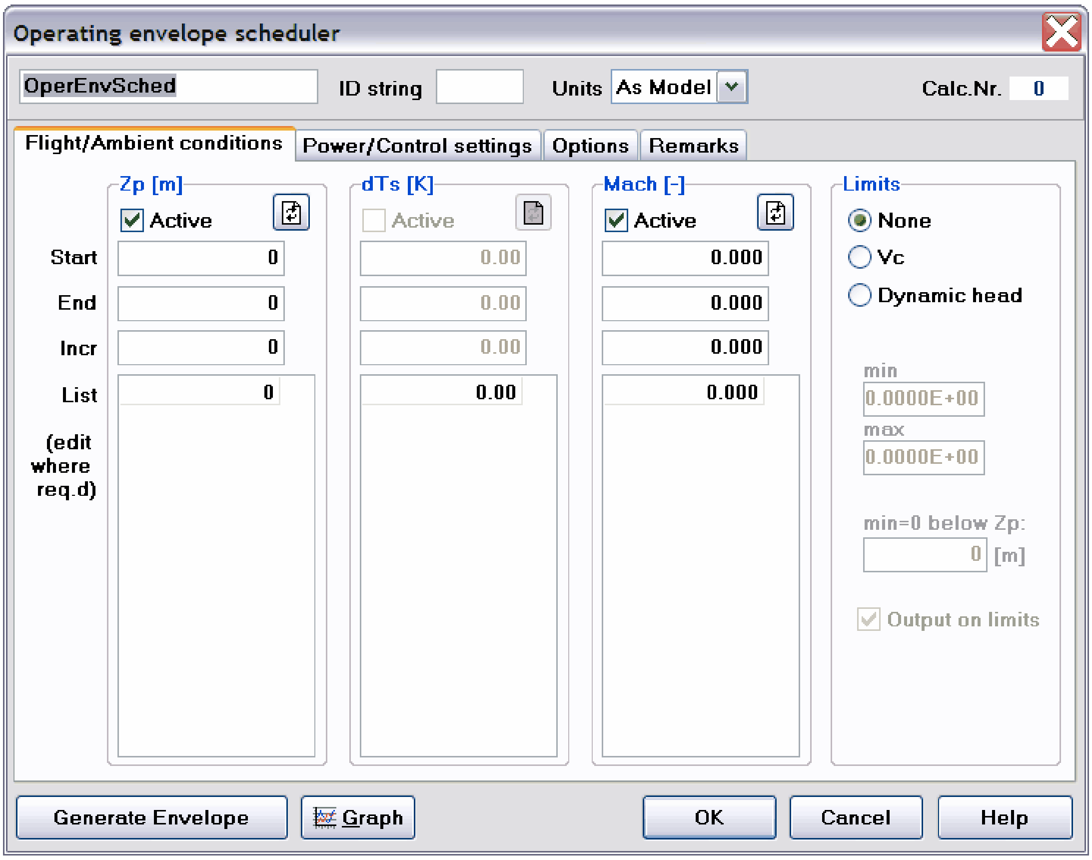
Quick start basics - Performance Deck Generation

Create a case (accept configuration changes) and name it "Performance Deck' and set the run case type to `St. St. Series`.

Drag a flight envelope scheduler component from the case component library to the newly created case `Performance_Deck`. The model window pane should resemble something like the figure beneath.

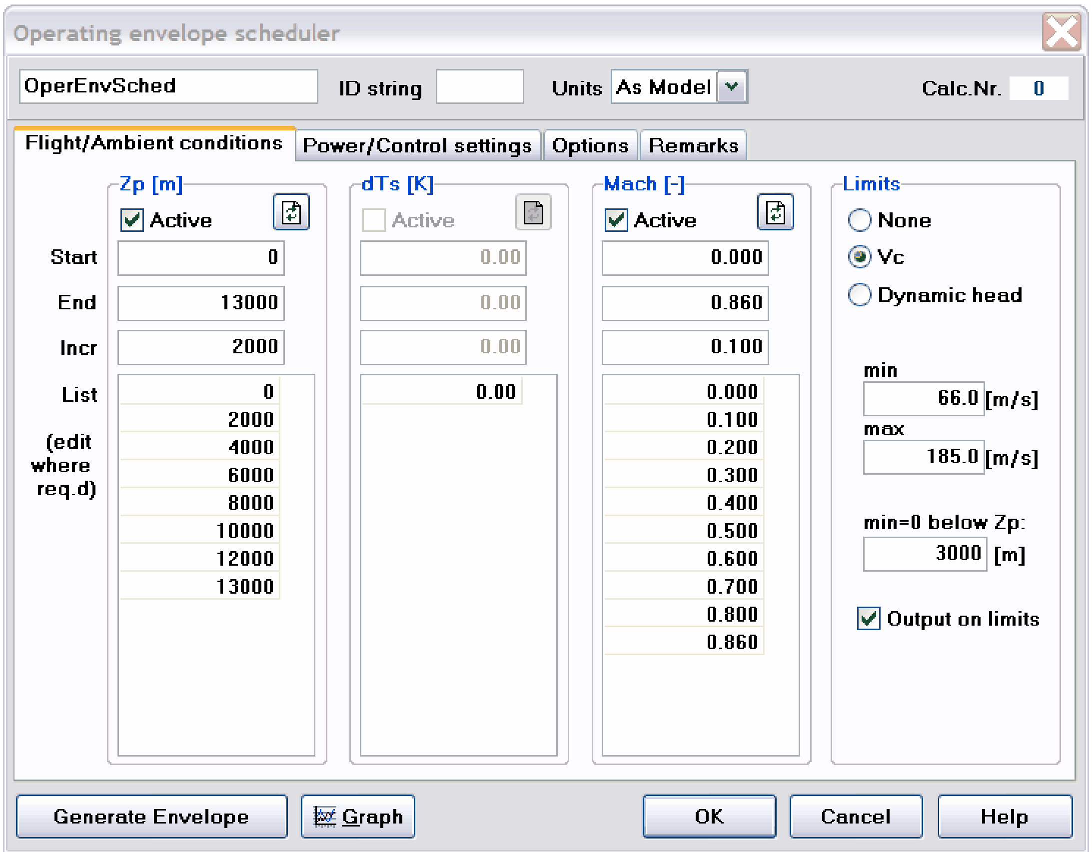


Open the data entry window of the flight envelope scheduler to configure the flight envelope. On opening the following window is presented:

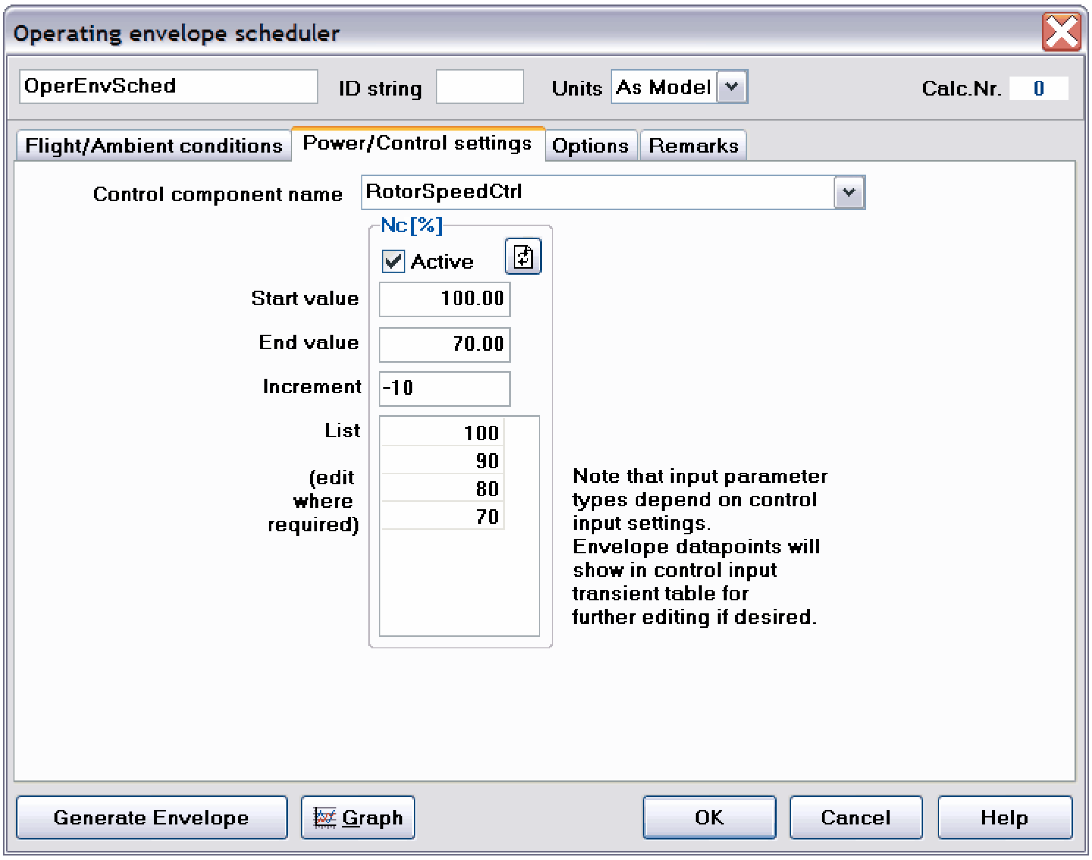


Basically three control parameters are available for creating the basic flight envelope. Note that the dTs parameter is disabled, but can be enabled by setting the Ambient/Flight conditions `model type` to `ISA+`.

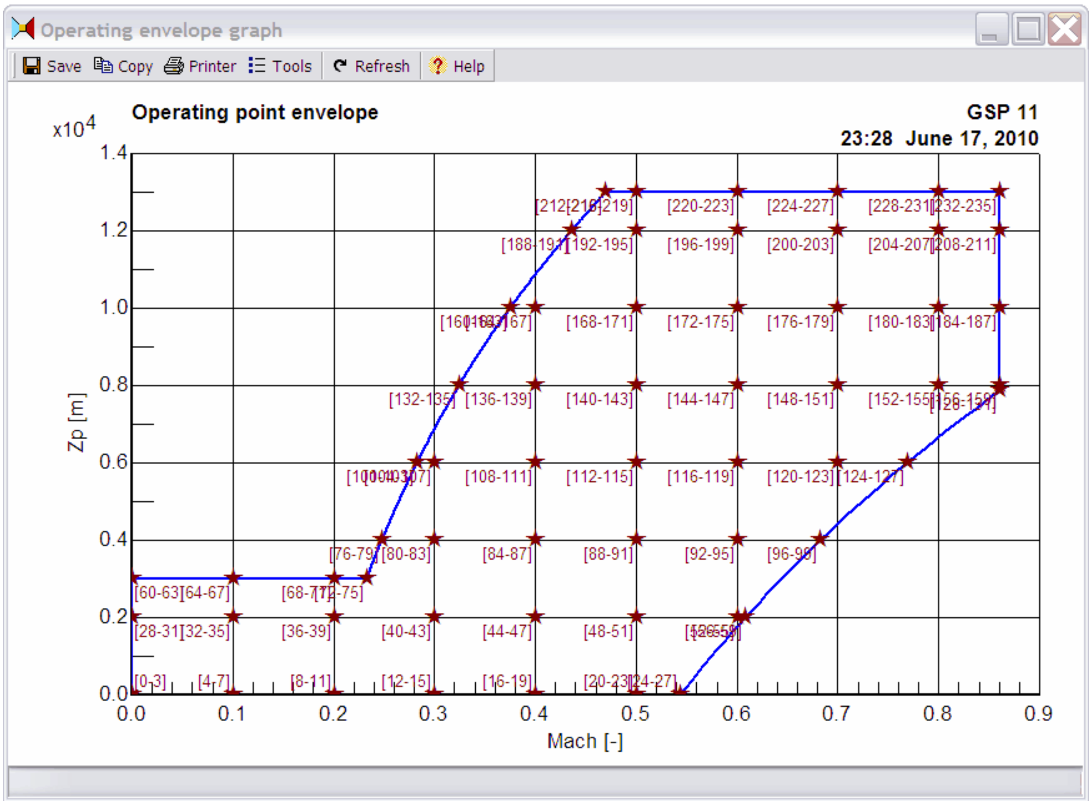
Setting a `Start`, `End` and `Incr` value for each of the flight envelope parameters will automatically create an incremented list from start to end. Pressing the insert key for a selected row in a list will insert a new row. Setting the limits will result in limiting output to area's the airplane is not capable of, or limited to fly. Change the input to the input listed in the figure below:



And the power control setting accordingly to the picture below:



Press `Graph` to generate the input data (accept `Apply Envelope`) which will open a graphical representation of the flight envelope. By generating the flight envelope, the appropriate input data will be generated in the `St.St. OD Series` input grid of the Ambient/Flight Conditions and in the `St.St. OD Series` input grid of the rotor speed controller.



Close the flight envelope graph and close the flight envelope scheduler. Since the simulation contains many points, the calculation time is considerable, and user interaction when errors occur is not preferred. By setting the `Options` -> `Transient/Series...` -> then by checking the `Batch Mode` option user interaction is not required.

Run the simulation to create the performance deck data.

6 GSP environment

6.1 Docking

In GSP 11 window docking is introduced. This allows windows (or rather window panels) to being docked to facilitate project configuration management.

There are 2 main docking window locations:

- Docking in the main application window:
  The main window has been extended with a 'docking site' allowing all the standard and custom component library windows to be docked inside the main window or in separate windows alone or together with other libraries. The docking style can be user adapted by selection from a number of predefined styles including the XP theme style that will reflect the Windows XP desktop theme setting. The main window docking layout and style can be saved, restored and reverted (StdModeldockingscheme.xml) to a standard layout if necessary. This facilitates the user to optimally configure his 'GSP desktop' only once. Please note that the main window should never be maximized on the screen to prevent the problem of not being able to drop the dragged component from library to the model/project window (because the main window hides the model/project window).

- Docking in the project window:
  All non-modal forms representing the project including the model configurations, display of results, reports or information, can be docked by the user to personalize the user interface. An interface to manage these window configurations is provided in the model window view menu to facilitate docking layout saving, restoring and reverting (StdModeldockingscheme.xml) to a standard layout if necessary. This facilitates the user to optimally configure his personal standard 'GSP model environment'.

6.2 Environment options

In the environment options window, a number of general (globally effective) options can be set:

`General` options:

- `Play intro sound on start-up (707 take-off)`
  When GSP starts a jet engine sound is played. Checking this option enables this sound.

`Graphical output`

- `Black and white printing`
  This forces the printer to print black and white (no grey-scaling) in order to avoid nearly invisible lines due to grey-scaled coloring.
- `Use enhanced metafile format (*.EMF) as default`
  This sets the default metafile format for exporting GSP graphical output, instead of standard *.WMF format.

`Logging and Auditing`

- `GSP user name`
  Enter a username for the logging and auditing the users actions of the GSP session. Note that this can be the name of the user that uses GSP, not necessarily the registrant! THis name is mainly used to display which changes are made by which user in the data logging window.
- `GSP user ID`
  Enter an optional ID for logging and auditing the users actions of the GSP session.
- `Confirm GSP User data at program start-up`
  When enabled, before loading the GSP main window users must fill in their username and user ID. The dialog defaults to the user name and ID that are stored in the GSP user name and ID fields. On cancel, GSP will terminate.

`Latest GSP version`

- `Check for update`
  Enable to allow to check for a newer version.
- `Update frequency`
  Select the frequency at which GSP checks whether a new version is available. Options are: at `startup`, `weekly`, or `monthly`.
- Optionally press the `Check now for update` to instantly check if an update is available.

`Warnings`

- `Warn for deprecated components`
  Some components are deprecated and can be replaced by more powerful components that can do even more. Example, the thrust controller is deprecated as the same can be done with the equation control components (you can even base the thrust on a map or a table).

`Files and locations` options

`Projects`

- `Warn when upgrading old version models during loading`
  Newer GSP versions loading older models may need to modify model files. Although this usually does not cause problems, a warning can be generated if desired.
- `Start-up model directory`
  The default directory location can be set here for the open dialog for browsing a model or project file.
- `Use last opened model file path for next start-up`
  Remember the directory on closing the last model or project file. When opening an existing model or project the stored directory will be the initial directory for the open file/project dialog.
- `Delete invalid entries of the reopened model and table files`
  When enabled, the non-existing files in the reopen file lists (re-open model or table) will not be deleted. If unchecked (default), files will be shown as disabled file links (cannot be selected).
- `Check for sample maps and models in user documents folder`
  During installation the smaple projects and maps are stored in the GSP program files folder. These projects cannot be saved by non-administrator users, this option will check if the sample projects and maps folder is present in the user documents folder when browsing for (open) a project.

The following is applicable when case management features are enabled (see option `Use case management features` below).

`Session Auditing` (Notice: Session options are effective in the next GSP session, this requires a restart of the application)

- `Generate session audit text file`; Enable or disabled the gathering of session audit data
- `Unique session file name`; Create a unique ID for the session file name (based on the date)
- `Save session in alternative directory`; Save the session data file in a different location other than the application data folder (`AppData`).

`Auto save` options

- `Enable`
  Enable or disable auto saving
- `Save interval in ... minutes`
  Define the interval of auto-saving the project data
- `Save project in alternative directory`
  Check to save in an alternative location, default folder is the users NLR/GSP `AppData` folder (Windows default).

`Case Management` options

- `Use case management features`
  Enable case management features (project versus single model file)
- `Access options`
  These options allow access and input that may conflict with the standard Case management rules. Use these options in case you want to fix or improve existing

project tree layouts and if you are an advanced user / know what you are doing. All access options are inactive per default.

- o `Allow case input in configurations`
  Check this option if you wish to make configurations with pre-configured case (design/off-design) inputs.
- o `Allow OD input in DP cases`
  Check this if Off-design input is to be allowed in Design point cases
- o `Edit components without case type selected`
  This option allows you to open and edit the component data entry windows before a case type has been selected. Only useful under exceptional conditions.
- o `Allow copy-paste nodes outside parent sub-tree`
  When pasting a tree node copy, only the data different from parent are copied due to the inheritance mechanism, so when a different node becomes parent, results may be unexpected and chaotic unless you know what you are doing.
- o `Allow copy case or config to parent`
  With this option active, the 'Copy to parent' submenu item (right click on node) becomes visible and all data in the selected node will be copied to the parent. As a result the parent will become identical to the selected child node and also other child nodes of the parent may well be affected through inheritance.
  - With the '`Allow copy case to parent configuration`' option active, cases can also be copied into configuration parents (otherwise the 'Copy to parent' submenu item is disabled) . This option can only be selected when also the `Allow case input in configurations` option is active. It is to be used with care since case data will be copied in configurations where they normally cannot be entered and may affect other child nodes of the parent node.
- o `Limit node depth`
  Node depth is limited by default for model load performance. Uncheck to have unlimited model configuration freedom.
- o `Limit config/case name length`
  Configuration and case name are limited by default to 10 characters. Model notes can be used for more extensive descriptions. Uncheck to have the freedom to longer configuration and case names.

- `Save options`
  - o `Delete unused series/transient input data`
    With this option active, series and transient input data are deleted if GSP finds they are not used, for example if case type is changed to single point cases (Design point, Steady-state). The user is asked for confirmation. This option is active per default.
- `Log data handling upon Revert Config/Case to parent`

`Security markings` options

This tab sheet manages 3 databases to set security strings for

- `Classifications`; Create/modify a user defined table with classification words/phrases.
- `Codewords`; Create/modify a user defined table with codewords/phrases.
- `Caveats`; Create/modify a user defined table with warning words/phrases.

`Enforce security markings setting in new model`

When enabled, on creating a new model, a dialog to set the security markings will be shown. There is no way to circumvent this dialog. On cancel no new model will be generated. This will enforce a compulsory security marking for a model.

`Project defaults`

This tab sheet is to specify default values for options in new created projects. These may include Project options (for the entire project including all configurations and cases in the project) and the various options groups applying per individual configuration or case.

- `General options / Initialization`
  - o `Calculate map scaling factors in Design Point cases`

`Graph templates`

This tab sheet contains a table with an interface to store frequently used graphs, e.g. after a simulation a graph of the output dat FN (thrust) and TSFC (thrust specific fuel consumption) may be shown as function of the POINTTIME (simulation step) or corrected spool speed, if configured here, this template can be easily selected after the simulation from the graph window.

6.3 Starting GSP

Start GSP by using `Start|Programs|NLR GSP 12|GSP 12` or by double-clicking the GSP icon if you have made a shortcut. At program start-up the GSP the splash screen appears, next the main window appears.

You can also open GSP bij double clicking on a GSP project file from a file explorer program, but beware not to open multiple instances of GSP as you will run into conflicts regarding database access.

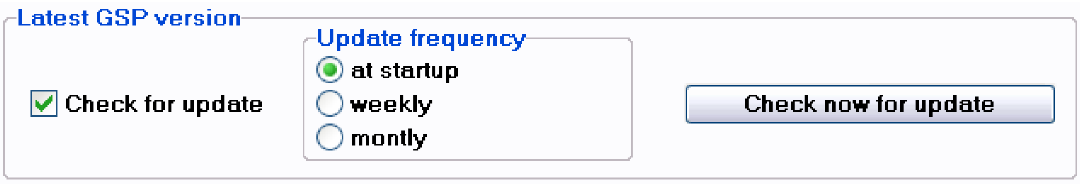
6.3.1 Splash screen

The splash screen is the window that is displayed when the GSP application is loading. Depending on the `intro sound` option of the environment options a sound representing a Boeing 707 taking off will be heard while loading the main application. The splash image will show the major release number.

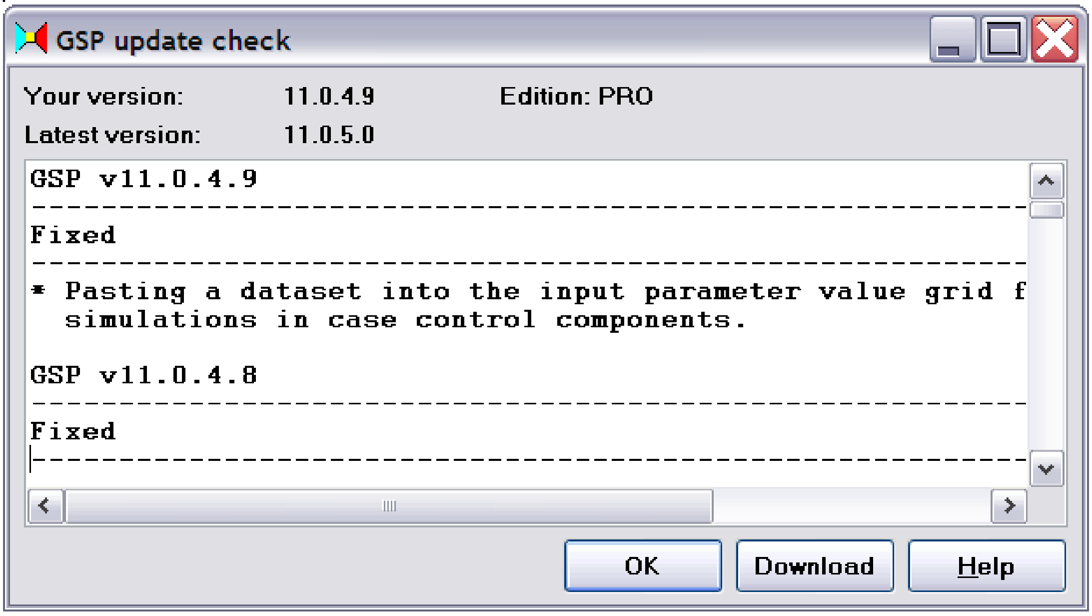
6.3.2 Update window

To automatically check for updates, the `Check for update` option on the tab sheet `General` of the environment options needs to be enabled. If enabled the frequency can be set. The program will check if an update is available, by default this functionality is enabled and check at every startup if a new version is available. Disable the option when no internet connection is available, or strict firewall rules are applied.

Environment options:



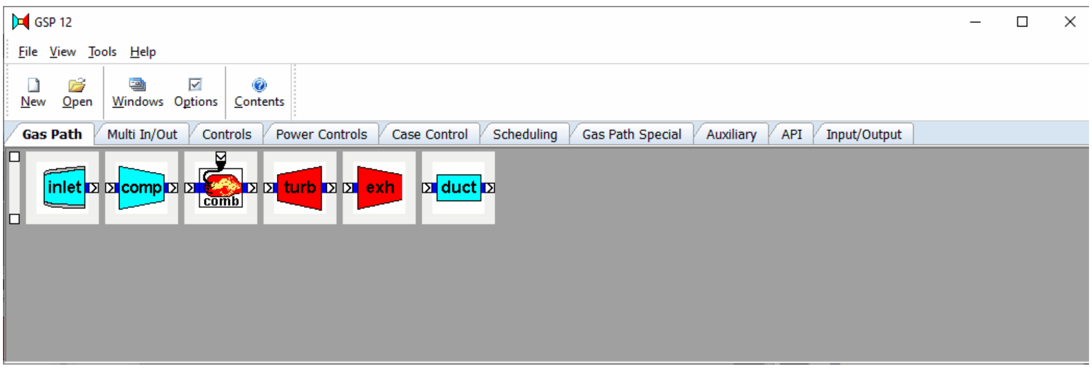
If an update is available, the update window will be displayed. Press the download button to proceed to the download area on the website.



6.3.3 Main window

The main window facilitates all general environment tasks - create new models/projects, open existing models/projects, open existing result tables, manages GSP windows, set general options and access online help. A docking region has been created to facilitate the docking of the component library windows.

Please note that the main window should **never** be maximized on the screen to prevent the problem of not being able to drop the dragged component from library to the model/project window (because the main window hides the model/project window).



On startup library windows are visible and docked (or invisible and floating, depending on the last state GSP has been exited). With respect to previous versions the library windows now have been categorized in to multiple library windows to facilitate easy browsing through/quick finding of the components.

Both the GSP LE (Light Edition) and the registered version contain component libraries spread over multiple dockable component library windows. However LE edition users can edit/view registered components, but are prohibited to run models with these components. Viewing and editing of these components has been added to evaluate the registered components. More information on the different libraries can be found here.

The main application window contains the Component Libraries, where components are categorized into separate tabs.

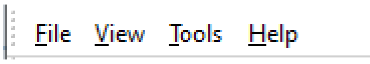
Main window buttons:

- The `New` and `Open` buttons on the main window create new projects or open existing models or projects. Multiple models can be open simultaneously. With disabled case management features these buttons are named '`New Mdl`' and '`Open Mdl`' respectively.
- The `Table` button enables access to result tables generated and saved in previous GSP sessions.
- The `W. List` button reports all open GSP windows.
- The `Options` and `Contents` buttons open the environment options and help menu respectively.

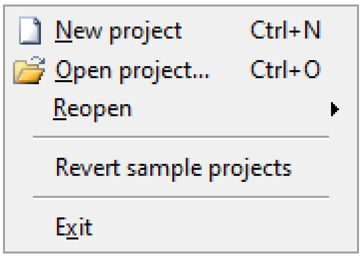
The menus and buttons offer a number of extra options and settings.

6.3.3.1 Main window menus

The main window menu toolbar:

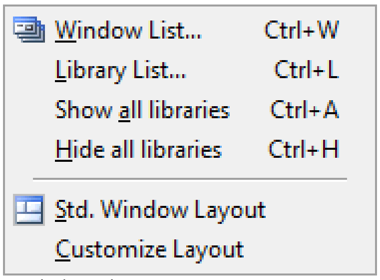


`File`



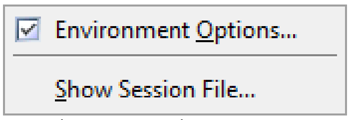
- `New project...`
  Create a new project/model.
- `Open project...`
  Open an existing project/model.
- `Reopen`
  Reopen a previously opened and closed model.
- `Exit`
  Exit GSP.

`View`



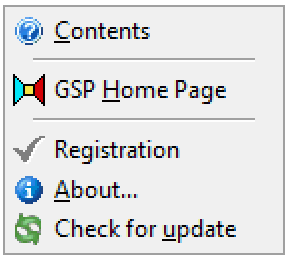
- `Window List...`
  Opens a dialog with list of all (main) windows
- `Library List...`
  Display list of all library windows
- `Show all libraries`
  Make all library windows visible
- `Hide all libraries`
  Make all library windows invisible
- `Std. Window Layout`
  Revert to standard window layout
- `Customize Layout`
  Customize docking layout

`Tools`



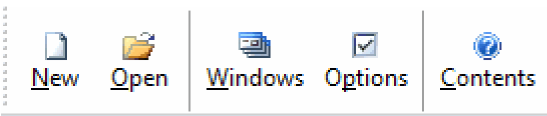
- `Environment Options...`
  Show the Environment options
- `Show Session File...`
  The GSP session file will be launched in the default text editor. Note that session editing must be enabled from the Environment options.

`Help`



- `Contents`
  Opens the contents of the help file.
- `GSP Home Page`
  Launches the GSP home page in the default browser.
- `Registration`
  Opens the Registration window to register the application
- `About`
  Shows the about box displaying e.g. build version information.
- `Check for upgrade`
  Shows the registration window for (un)registering GSP

6.3.3.2 Main window toolbar



`New`

Clicking this button creates a new model file.

Open

Clicking the highlighted button brings up an open file dialog for opening existing model files.

Windows

Clicking the highlighted button (or ctrl-w) brings up a list with open GSP windows. This is convenient to find a window hidden behind a lot of other windows on the desktop.

Options

Click the highlighted button to open the environment options window.

Contents

Click the highlighted to go to contents window of this on-line help guide.

6.4 GSP Projects

6.4.1 Introduction

GSP 11 has introduced a set of integral engine model data management features named "Case and configuration management" or just "Case Management". The objective of this functionality is to improve usability, productivity, data integrity, and security of gas turbine model data. Different model configurations and simulation run cases can be defined in a single GSP project and saved in a GSP project file*. The configuration, case and model data in project file are all stored in xml format. As such, GSP projects may range from a single gas turbine model and run case such as for example a single design point up to complex projects with multiple different model configurations each of which with several design point, off-design (both steady-state and transient) simulation run cases.

GSP 12 takes the data management features even further by incorporation of a modern database SQL based data storage system which allows output data to be managed better and compared between the various run cases.

An important element in a GSP project is the use of object orientation and inheritance. Configurations and cases are ordered in a hierarchical manner to minimize data duplication and facilitate easy management of differences in model configurations and simulation input data.

* In GSP 10 (and older versions) individual gas turbine model configurations could be made and saved in model files in binary formats. With this single model configuration different types of run cases could be defined but not individually saved in a single file. This meant that for different cases, several different model files had to be saved resulting in elaborate file management tasks once projects got more complex.

6.4.2 Importing older models

Up to GSP 10, GSP model data were stored in binary format in .MDL files corresponding to only a single configuration and case combined. Loading these old model files into GSP 11 has some conversion issues.

Usually GSP automatically upgrades your old model to the current version since GSP is backwards compatible. The .MDL file data are loaded in the Base model configuration of a new created project with a corresponding project file name.
However, several settings in the model may need attention:

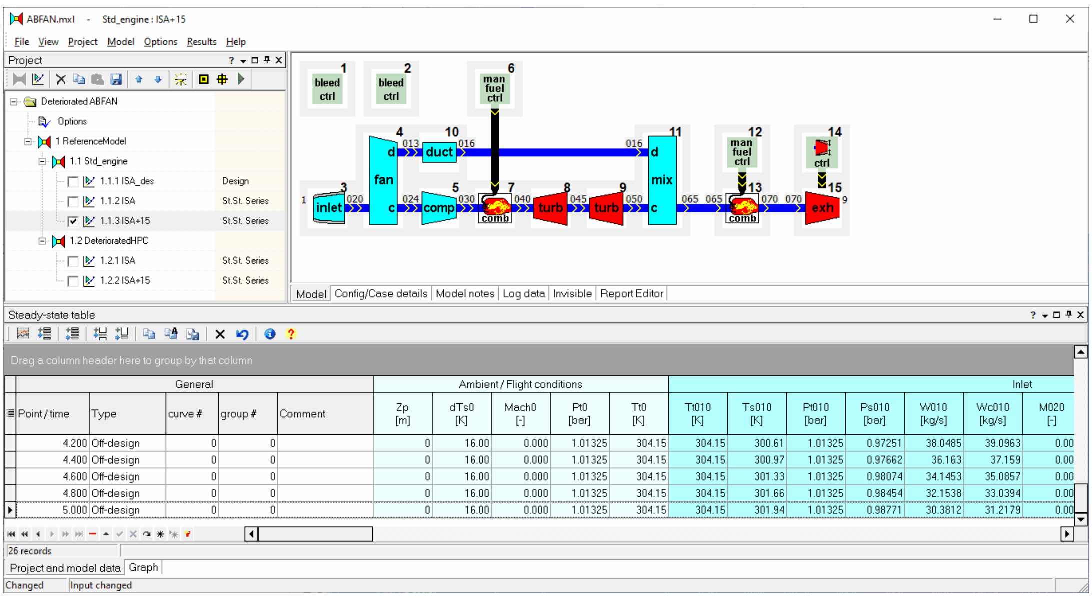
- Duplicate station numbers
  No duplicate station numbers are allowed in GSP 11 and up. Station numbers in between components or inside components (e.g. the turbine NGV station, found on the output tab

sheet of the model's input window) are checked for duplicates (on initialization and on calculation). Please check and correct duplicate station numbers.

- Case type components in configuration type
  The Reference mode node in which the old model file data are loaded, is of type configuration, and restricts usage of gas path components and control components (basically the definition of the mathematical engine model). Input modifiers as case components, or any component that acts as input specifiers for the model are solely allowed to be used in case nodes. Upon loading older model a confirmation message dialog will be displayed when an input specifying component is placed in the base configuration model node. The modeler can choose to delete this component immediately, or leave it in the configuration. If the modeler chooses to leave it in the configuration, the modeler should be aware that the component must be moved to a case (first create a case child node to the base configuration model to paste the component).
  The problem can quickly be corrected by right-clicking the component and selecting the `Move component to child configs/cases` menu item to move the component to a child case. If there is no child case it must be first added or inserted, for example using the `Insert Child Case or Config menu item` when right clicking the selected configuration node.
- Duplicate output parameters
  Duplicate output parameters are not allowed. The new version checks the output parameters on duplicates during initialization or upon running. Please remove the check marks of selected duplicate output parameters that can be found in multiple components. E.g. the spool speed can be specified in any component that is attached to the same shaft, and hence may result in duplicate output parameters. A quick solution is to use the centralized component (`output parameters` tab sheet) output to remove the duplicate selections. When selecting the spool speed for option `all turbo components` automatically configures the components that are attached to the same shaft so that no duplicate output is obtained.

6.4.3 Project window

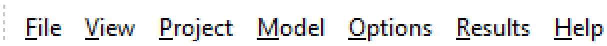
The project model window offers a well-organized presentation of a series of gas turbine models (configurations and cases). On the model panel, components are arranged and linked in a configuration similar to a specific gas turbine. Depicted below is an example of a typical project window including the basic elements of a complete GSP gas turbine model. Click on the various elements in the image to learn more about how to use the model window.



The project model window features docking. Docking enables the user to place the windows according to his/her preference/convenience.

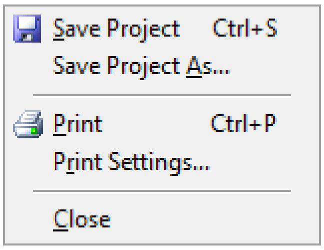
6.4.3.1 Project window menus

The model window menu toolbar:



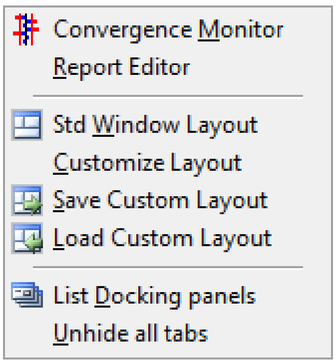
Please note, the options of some of the menus displayed below are depending on the model component (and type) and its place in the inheritance (config or case model). Some options may or may not be enabled (disabled is grayed out) when you open the menu. The reason is that the option/operation is not applicable, you need to select a component or change the model node for instance.

`File` menu



- `Save Project`
  Saves the project.
- `Save Project As...`
  Saves the project with a different name.
- `Print`
  Print the current model.
- `Print Settings...`
  Change printer settings.
- `Close`
  Close the current model.

`View` menu

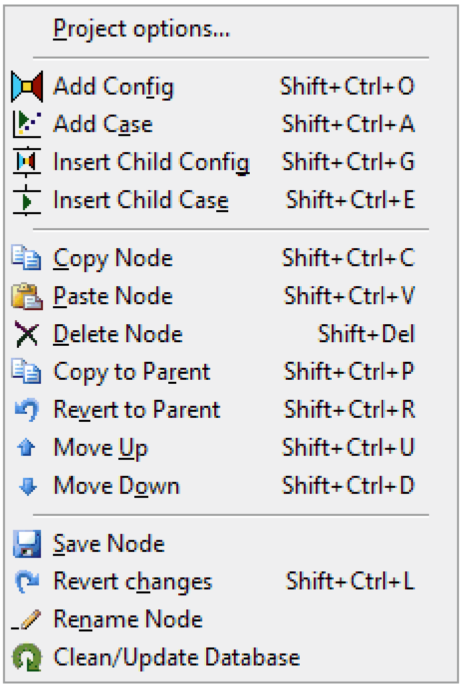


- `Convergence Monitor`
  The Convergence Monitor graphically displays the error evaluation progress during simulations.

    - `Report Editor`
      The Report Editor is a simple notepad like editor which is used by GSP to display error reports or model performance reports for instance.
    - `Std Window Layout`
      Restore the model is to the default state when GSP was originally installed.
    - `Customize Layout`
      Change the appearance of the windows
    - `Save Custom Layout`
      Save the customized window layout.
    - `Load Custom Layout`
      Load a customized window layout.
    - `List Docking Panels`
      Show a list with all windows and their state (docked, undocked or hidden), double click to activate the window.
- `Unhide all tabs`
  Shows all windows (docked, undocked or hidden).

`Project` menu

Note that some items in the menu can be inactive, depending on the type of the tree view node, some actions are enabled and some disabled. E.g. only cases are allowed to be run-able, therefore disabled in all nodes except for the case node.

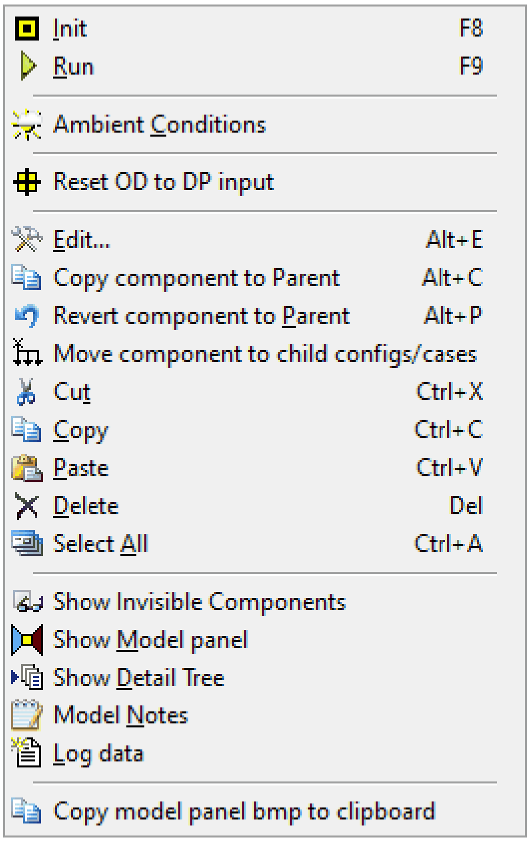


- `Add Config`
  Add a `Configuration` model as a child of the current selected node. Note that when child nodes exist, the new node will be added below the last child node.
- `Add Case`
  Add a `Case` model as a child of the current selected node. Note that when child nodes exist, the new node will be added below the last child node.
- `Insert Child Config`
  Insert a `Configuration` model in the parent of the current selected node, where the selected node (and its children) will be the child of the inserted node.
- `Insert Child Case`
  Insert a `Case` model in the parent of the current selected node, where the selected node (and its children) will be the child of the inserted node. Note that a `Case` model cannot be inserted when the selected node is of type `Configuration` (option is then disabled)

- `Copy Node`
  A complete copy of the node and its children will be made.
- `Paste Node`
  After pasting the copy somewhere in the configuration, the inheritance functionality will remove duplicate data, so that differences with respect to its ancestor are seen in the details tree view. Pasting nodes is allowed to the parent of the copied node, or as a child of the copied node.
- `Delete Node`
  Delete the selected node and its children.
- `Copy to parent`
  Copy all case/config model data to the parent case or config. This is convenient if after some editing one realizes the changes should have been entered in the model data of the parent or an ancestor case or configuration.
- `Revert to Parent`
  Re-inherit all parent model data. This is convenient if after some editing in a case for example one wants to revert all changes, i.e. reset all model data parent case/config.
- `Move Up`
  Move the node up within the current parent of the node.
- `Move Down`
  Move the node up within the current parent of the node.

- `Save Node`
  Save the current changes of the configuration.
- Revert changes
  Revert the changes to the stored model (revert all model changes to state as when last loaded from current case/config)
- `Rename Node`
  Rename the configuration.

`Model` menu



- `Init`
  Initialize the model.
- `Run`
  Start the simulation if a valid case type is selected.

- `Ambient Conditions`
  Edit ambient/flight conditions.

- `Reset OD input to DP values`
  Reset all numeric OD input to the respective DP values.

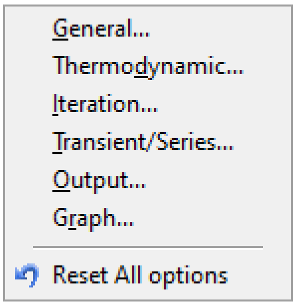
- `Edit...`
  Opens the Component data window
- `Copy component to parent`
  Copy the complete component model to the parent node (base config, config or case node)
- `Revert component to parent`
  Reverts all the input data to the parent component
- `Move component to child configs/cases`
  Move the instantiation of this component to child node(s) (configurations and cases)
- `Cut`
  Cut the selected model component(s).
- `Copy`
  Copy the selected model component(s).
- `Paste`
  Paste the selected model component(s).
- `Delete`
  Delete the selected model component(s).

- `Select All`
  Select all model components

- `Show Invisible Components`
  Opens a window and displays the invisible components in lists (shafts, bleed inflow and bleed outflow)
- `Show Model Panel`
  Show the model panel/window containing the stacked components representing the gasturbine model.
- `Show Detail Tree`
  Show the window containing the details of the gasturbine model (differences with respect to the ancestor).
- `Model Notes`
  Show the model notes window for this model.
- `Log data`
  Show the window containing the log messages of the major actions performed by the user (auditing).

- `Copy model panel bmp to clipboard`
  Copies the current model panel to clipboard to paste in e.g. reports.

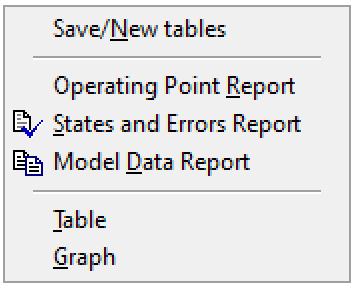
`Options` menu



- `General`
  Specify general options like enabling confirmation dialogs.
- `Thermodynamic`
  Specify thermodynamic gas model options like real gas corrections.
- `Iteration`
  Specify iteration specific options like accuracy and numerical settings.
- `Transient/Series`
  Specify transient or steady state control parameters like start time and timestep.
- `Output`
  Specify output options like automatic table pop-up and gas composition output.
- `Graph`
  Specify graph options for the Steady State results graph window like scaling and titles.

- `Reset All options`
  This will revert all the options to the initial default value.

`Results` menu



- `Save/New tables`
  Optionally save data in current outputted steady state and transient table data, and reinitialize tables based on all output parameters selected in the Output tab in the component data window of the components.

- `Operating point report`
  Generate a text report for the last calculated (steady state or transient) operating point in the GSP text report window based on the output parameters selected in the Output tab in the component data window of the components.
- `States and Errors Report`
  Generate a text report for an overview of the current states and errors in the GSP text report window based on the model configuration (model initialization is required, requiring a valid linked gas turbine model).
- `Model Data Report`
  Generate a text report for an overview of the current input of the model components in the GSP text report window based on the model configuration.

- `Table`
  Display the created table based on the calculated data
- `Graph`
  Display the created transient graph based on the calculated data, if no data in the table, the graph will not be presented.

`Help` menu

- `Contents`
  Display the help file contents
- About
  Show the GSP about box to display e.g. build version information.

6.4.3.2 Project Options

To access project options now single click the options node of the project tree. The project options window provides control over the following GSP project options:

`General Options`

`Project name`

A project name can be entered used for GSP output document identification. The project name also shows in the root node of the project tree.

`Project tree panel` options

- `Show node index`
  Create a node index for every node entry in the Project tree panel. The index string starts counting at the base configuration root model and index is based on the position of the node with respect to its parent; the levels are separated by a dot (e.g. 1.3.2 = the second case/configuration from the third case/configuration of the base configuration model node).

`Autosave` options

- `Autosave configurations upon exit`
  Do not ask for confirmation to save configuration data when exiting the configuration node in the case management treeview, but save always. Note that the only way to revert changes then is to not save and then reload the project.
- `Autosave case upon exit`
  Do not ask for confirmation to save case data when exiting the case node in the case management treeview, but save always. Note that the only way to revert changes then is to not save and then reload the project.

`Change propagation`

- `Propagate changes to child configs and cases`
  Do not ask for confirmation to save case data when exiting the case node in the case management treeview, but save always. Note that the only way to revert changes then is to not save and then reload the project.
- `Confirm changes`
  Confirm changes propagated into child configurations and cases with modified data (different than the ancestor being modified).

`Map data handling`

- `Warn on inconsistent map data`
  Warn when reading inconsistent map data such as column/row input values not the same in different cross tables in the same map, which for standard maps is required in GSP.

`Output tables`

Always show the case name column, also if only a single case selected; when inactive, only with multiple cases in the table the case name is shown to distinguish data from different cases.

`Advanced options`

`Show advanced model equation controls options in components`

With this option active, advanced model error equation and state create control options will be visible.

`Security options`

`Encryption`

- `Encrypt on saving project`
  On saving project the entire project file will be written encrypted into a GSP XML project file. Default, a password will be prompted (for non default behavior see `Use default password` below). A user defined password will be hashed using a 256-bits Hash algorithm (one way only!) after which the actual encryption algorithm (256 bits) encrypts the project.
- `Use default password`
  This enables the use of a default password by setting a user specific password. Use `set` to set a password that will be written encrypted to the windows registry. With `Encrypt on saving project` enabled the user will not be prompted to enter a password, **note that there is no way to recover saved encrypted files when the password is forgotten!**. Use `unset` to clear the stored password. Use this option wisely. A label text in bright blue marks the status of a set password.

The `Clear` button will clear the security markings, when completely empty, no markings will be shown in the project window.

`Security markings`

Security markings can be selected for the project. Note that the lists are defined in the Environment Options|Security markings.

6.4.3.3 Project window panels

The following dockable panels are part of the project window:

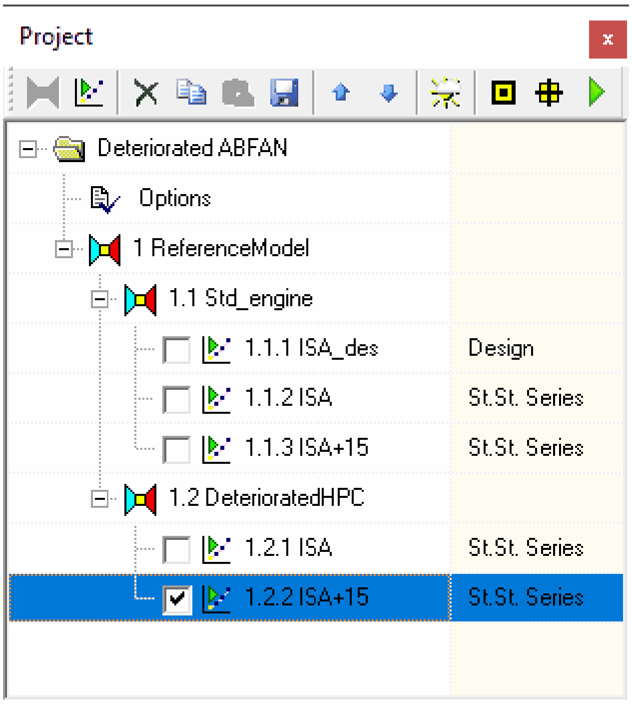
- Project tree panel
- Model panel
- Config/Case details panel
- Log data panel
- Output table panels
  - Steady state for DP and OD steady state output
  - Transient
- Output graph panels
  - Steady state for DP and OD steady state output
  - Transient
- Report Editor

Depending on the last state of the project window some panels may not be visible (hidden), to view these go to the `View` menu and select `List Docking panels` to unhide the hidden panels.

The users are free to position (drag and drop) the panels in any arrangement that suits there need for a comfortable user interface. Note that placement icons are displayed when docking of a panel is possible in another panel. Additional functionality is available to store a certain panel layout, or to restore the default (see the `View` menu of Project window menus).

6.4.3.3.1 Project tree panel

The GSP XML data structure corresponds to a GUI project tree with any number and level of sub-configurations and cases.



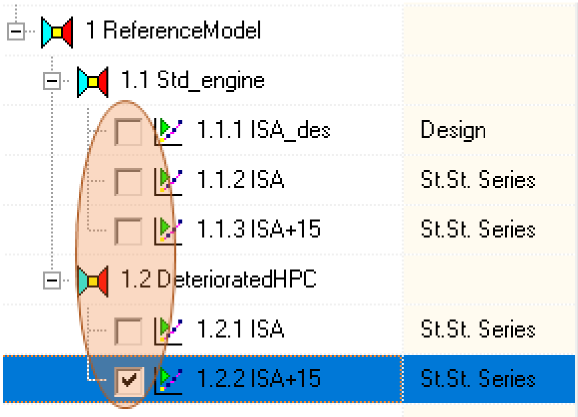
Project tree panel for case management configurations are particular arrangements representing a particular engine configuration. A configuration can have one or more cases that actually represent different input data sets such as operating conditions, power settings and/or control laws. A case can have any number of sub-cases but cannot have sub-configurations. Only cases can actually run simulations and produce output. In the

Config/Case details panel in the model window the case or configuration specific data are shown. Corresponding to the inheritance principle, these include all data that are different from the ancestor case or configuration. The Config/Case details panel facilitates rapid browsing through project trees to find particular case or configuration details.

A top configuration is represented by the base configuration (reference) model which usually is the engine configuration from which a design or performance analysis study is started. There is no restriction in the model arrangements in the configurations derived from the base configuration model. An extreme example is an empty base configuration model with a number of different sub-configurations.

Configure the tree using the menu, toolbar or popup menu.

GSP 12 introduces a check box in front of the case model name, see the orange accentuated area below:

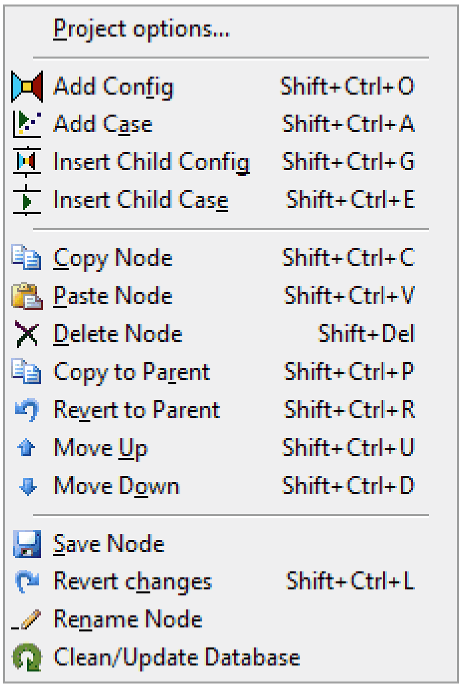


The dark blue background indicates that this is the current focussed/selected (case) model node, this selected node has automatically a check mark in the check box. By selecting other case check boxes, the selected model output table is joined with the (priorly generated simulation) data from the cases with a check mark so that simulation output runs can be compared.

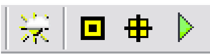
6.4.3.3.1.1 Using the project tree

There are 3 different ways to handle the configuration of the project. Note that some items/buttons are disabled (grayed out). Depending on which node is active in the project tree.

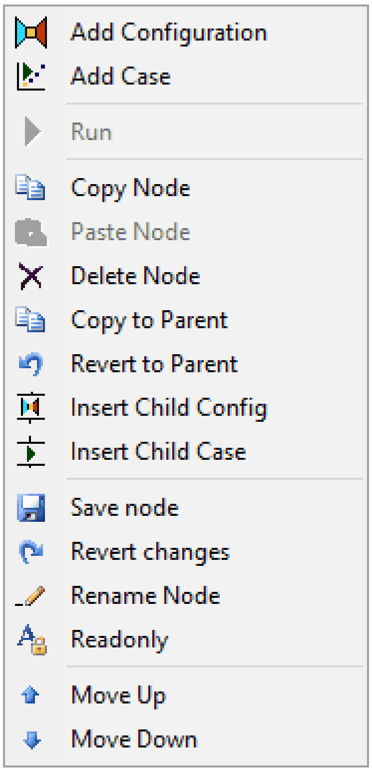
- Using the project menu of the model window



- Use the project toolbar on the Project tree panel window

This toolbar covers the basic functions described in the project menu. Additional buttons are: 
Flight/ambient conditions to edit de flight conditions, init button to initialize the model, the run DP and reset single point OD input to DP values button and the simulation start button (when green the button is enabled).

- Use right-click pop-up menu on the Project tree panel nodes (beware,depending on the position in the tree, or e.g. if a node is copied, it is allowed to insert a child config or case, etc. some options may be disabled because they are not applicable for the current selection; this implies that the menu may have less or more disabled/grayed out options than displayed below)



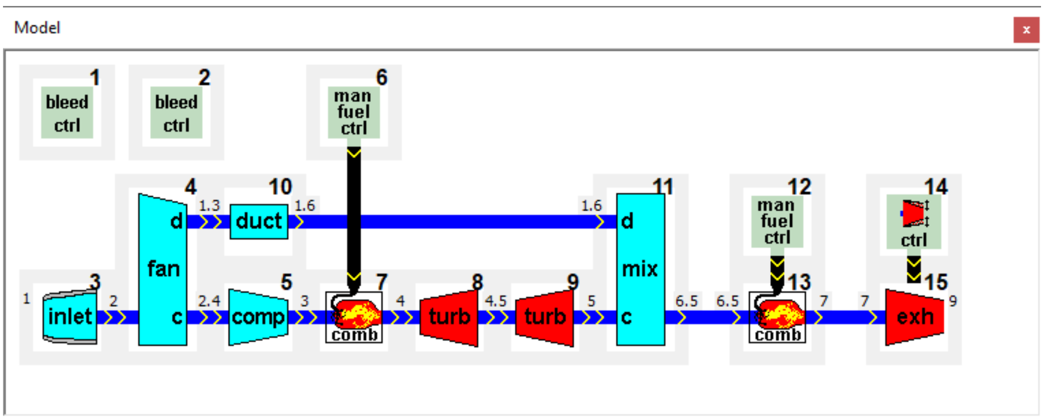
The different actions are described in project menu.

6.4.3.3.2 Model panel

The white area in the modeling panel is a scrollable region for containing the GSP component icons. The model panel holds the graphical representation of the selected configuration from the Project tree panel. Drag and drop model components from the Component Libraries to construct a model configuration.

Components can be dragged from the component libraries onto the worksheet of model window (note to not maximize the windows of either the GSP main application containing the component libraries nor the model window; with the GSP main window maximized you cannot access the model window!).

Undocked the window looks like:

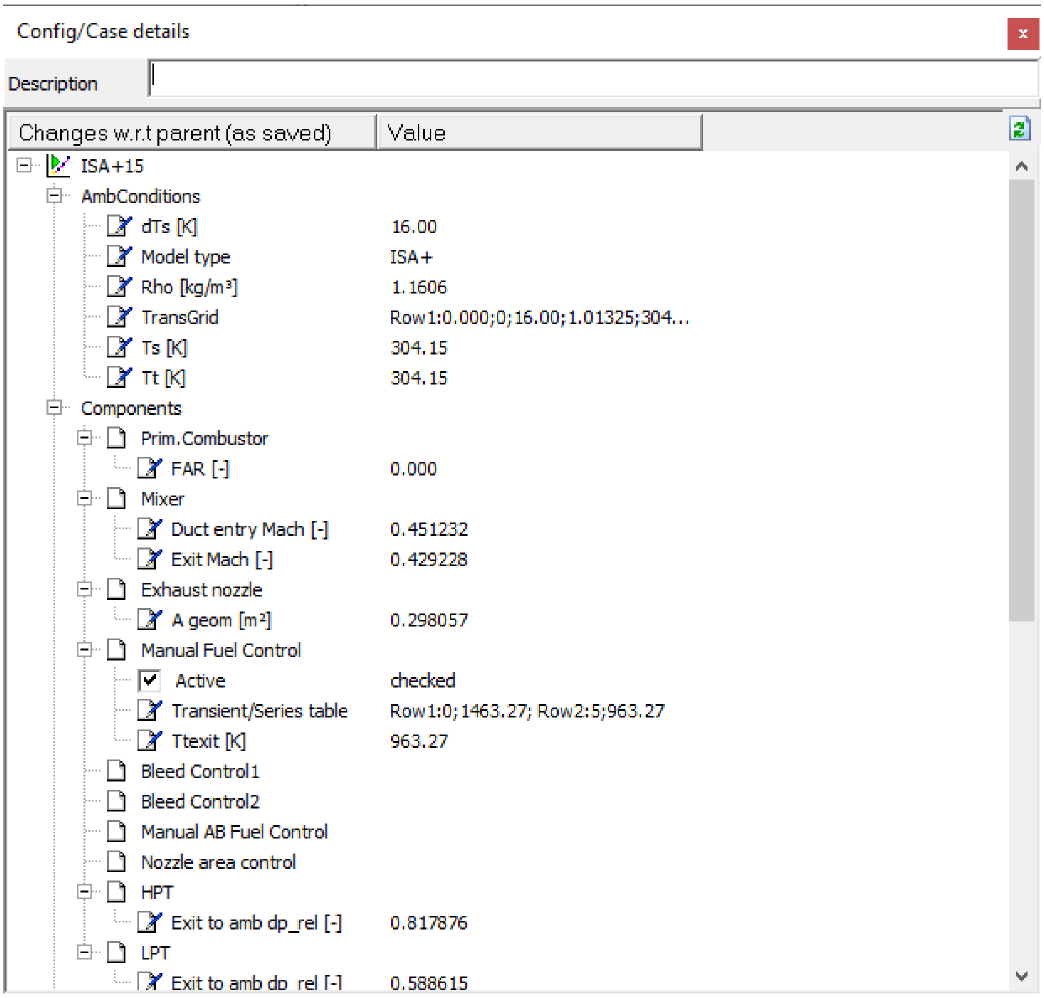


Default the window is docked with the Config/Case Detail view and the Log Data view.

6.4.3.3.3 Config/Case details panel

The config/case details view, as e.g. depicted below, shows the changed input data parameters with respect to the parent (as saved). This view enables quick reference to find the actual data that has been changed with respect to its direct parent.

By entering a short description on the model, a pop-up help tip will be displayed in the respective node in the Project tree panel window.



Using a double-click on either one of the columns the data property window of that component will open, and the selected control will receive the focus to enable instant editing.

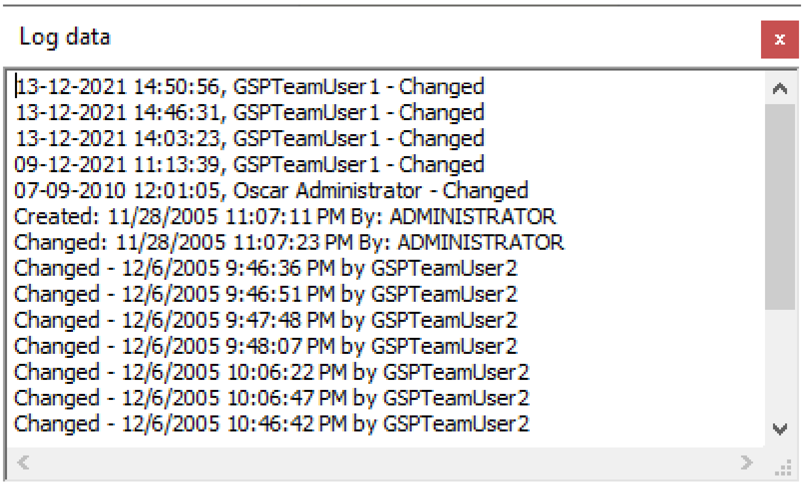
The 'Refresh detail tree' button found on the right of the table's column headers resets the detail data to the result of the difference between the saved data and the parent data. To refresh the tree a model node save is required (will be asked to save node when changes have been sumitted to the model).

Note that invisible controls (also if visible but on invisible tab sheets) are not shown. It parent nodes (e.g. components) have only invisible controls, then the parent node is shown but the actual invisible child control is not listed. This is to indicate the node (e.g. component) has data deviating from the parent case or config but these data are not active in the case.

Note that building the details data tree view requires significant processing power, especially with many changes present (or with all in case of the base configuration model). This means that for best performance when browsing the project tree, it is advised to select the mode panel (and not have the details panel visible).

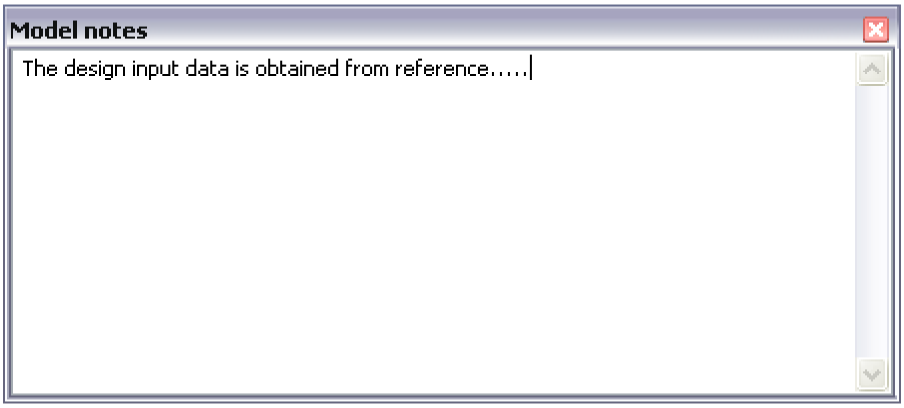
6.4.3.3.4 Log data panel

The log data view, as e.g. depicted below, shows the configuration specific logged events (audit data)to display the history of major changes. This is a dockable window dockable in the model window.



6.4.3.3.5 Model note panel

The model notes, as e.g. depicted below, shows the model specific notes written by the modeler. This is a dockable window dockable in the model window.



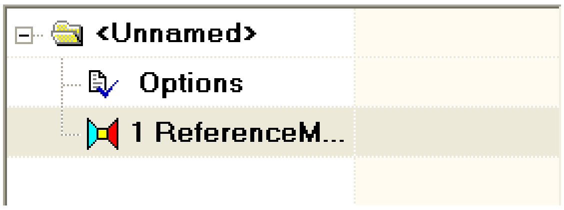
6.4.3.4 Creating a new project

The tree view (`Project`) of a new GSP project (or a converted *.MDL model file) will contain 3 nodes:

- A root node displaying a Project name specified in the Project options
- `Options` (clicking opens the Project Options window)
- `Base configuration` (Clicking activates the base configuration model configuration for editing in the Model panel).

On conversion of the old *.MDL file format the model will be copied in the base configuration model. Having more stringent rules of placing certain components in configuration/case levels, the modeler will be warned when a component is placed in the base configuration model on conversion that should not be there (e.g. case components). Just cut this/these components from the base configuration model and insert them in the appropriate configuration or case level.

The following figure shows the layout in the Project tree panel:



The project tree can be extended into a hierarchy of model configurations and simulation run cases following GSP's configuration and case management concept. This allows for rapid model changes to assist e.g. in the changes of requirements of the design process or the construction of a model of an existing changing engine.

6.4.3.5 Building models

A model of a gas turbine configuration is created by the GSP modeler by dragging the appropriate components and arranging these according to the layout of the actual engine of interest. A model of an engine generally is considered to be part of the engine project.

Elements of model construction are described in this section and it's paragraphs.

Note that since the introduction of configuration and case management, the input access rules require a strict model setup to ensure proper, useable models expressed in a project file. More of this is explained in Creating an engine project.

6.4.3.5.1 Creating an engine project

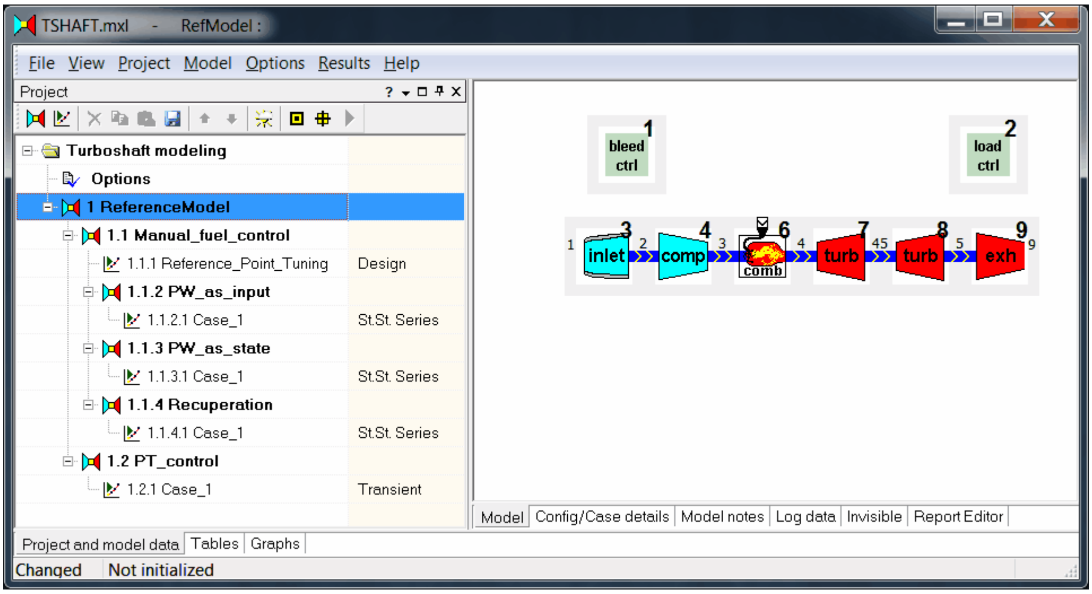
The creation of a well organized model ensures proper encapsulation of the modeling task. The creation of an engine layout is described in the Creating a model layout. This section will more closely describe the model design philosophy per the developers of the GSP gas turbine simulation tool.

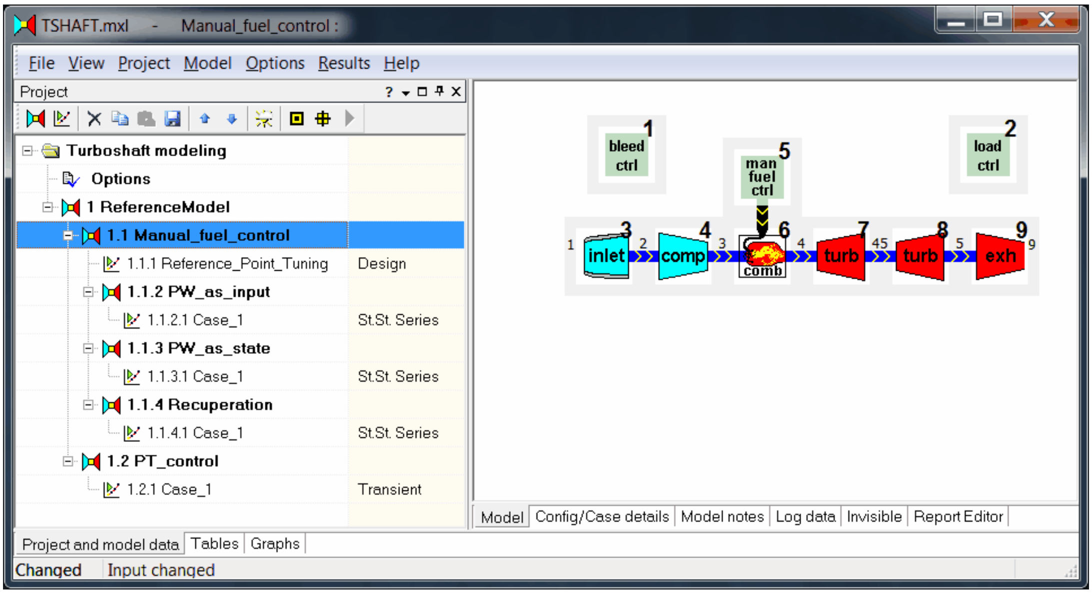
A base configuration or reference model is always the starting point for a model and should contain at least the components that are shared with all child configurations. This means for instance that an incomplete cycle can be modeled here (e.g. only gas path components and no fuel control components when the modeler is interested in e.g. manual fuel control and governor fuel control simulation). Note that the base configuration model can also be empty, this can be beneficial, but breaks with the inheritance approach.

A reference model should always be followed by a model configuration where the modeler completes the engine cycle (e.g. adding the fuel controller). To tune this model it is advised to always immediately create a design case model under this configuration to obtain a means to fine tune the created model configuration. Once the model design point (or better called, the reference point as the design of the actual engine not necessarily is the data the modeler has to create the model reference point; an engine design usually is a concession or trade-off between multiple design/important engine power settings and operational conditions) is fine-tuned, the model component input should be copied into the parent model configuration (and even further to the reference model).

Once the parent of the tuning case model (and even higher up the tree to the base configuration model) is updated, the model is ready for either sibling off-design case models (do not put these under your tuning case!) or child configurations as siblings to the tuning case model.

Please take a look below at the model setup of a turboshaft engine model where the base configuration model does not contain a fuel controller as the model is used for steady state analysis and transient analysis. Also note that the `Manual_fuel_control` configuration contains configurations that alter the model to either allow the user to input power (`PW_as_input`) or input fuel flow (`PW_as_state`) as power setting.





6.4.3.5.2 Creating a model layout

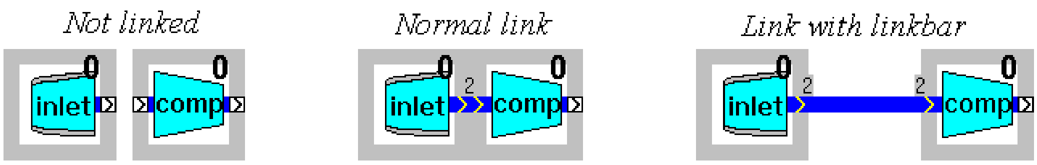
Building a GSP model is intended for the more experienced user. If you are a novice at using GSP, it is recommended that you start with running simulations with the existing sample models, as described in *Running a simulation*. Note that it usually is most practical to derive new models from existing or sample models. Open an existing model, adapt component data as described in *Entering component data*, and add or reconfigure components as necessary.

A new gas turbine engine model is built by configuring predefined generic components in a certain arrangement in a model panel using GSP's drag-and-drop interface. The components are dragged from component libraries onto a model window.
Use the `File|New` command from the Main window to open a new GSP project. In the new GSP project, go the the Reference model node in the project treeview to activate the Reference model configuration. On the right, an empty model window appears (if not, select the '`Model`' panel by clicking the '`Model`' tab or go to the `Model` menu and select `Show Model Panel`. Next, the gas turbine model is configured using simple drag-and-drop operations: select a component with the mouse from the standard component library, keep the mouse button pressed while dragging the component to the model window and release the mouse button to drop the component on the window. Drag more components from the libraries onto the window, and link them together using the link icons on the component icons.
Note that components can also be dragged, copied and moved to and from other existing model windows.

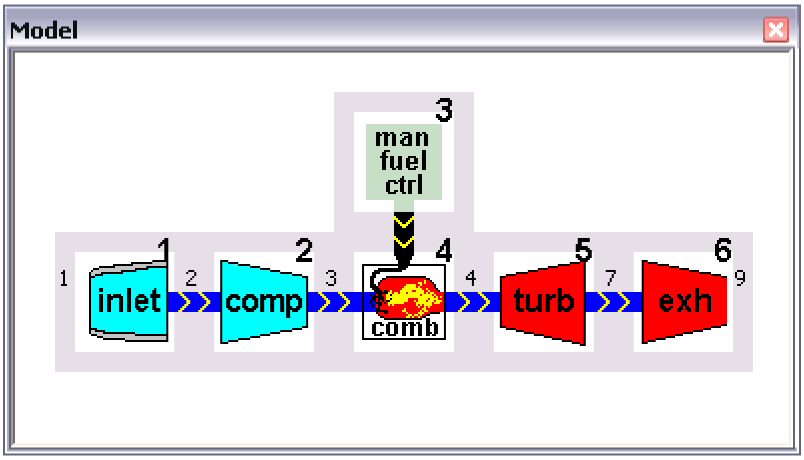
A particular gas turbine configuration is obtained by *linking* the components in the model window together in a corresponding arrangement. Components are linked by placing them adjacent to each other or using the link bar component. The links between components represent either

- gas path connections at user specified engine station numbers or,
- control inputs (such as a fuel flow signal from a fuel controller to a combustor component).

Other connections or relations among components such as shafts transmitting power or secondary air flows are specified using numbers in the component data and are not visible in the model window (see 5.2.1 and 5.2.2).
The chevrons on the link icons indicate the direction of gas flow or control input. GSP has defaults for specific (gas path) component entry and exit station numbers according to ARP 755A. When linking gas path links, GSP uses the default station number if both sides match (e.g. compressor exit and combustor entry station numbers are both 3), otherwise GSP requests the user to specify a number. Station numbers are also used to identify output parameters at particular engine stations.
Adapt the station numbers for the gas path links (click the link icons) if necessary. Fractional station numbers (e.g. 4.5 for between turbines) may be entered. If components are linked, the connection indicator turns dark blue for gas path links and black for control links, as shown in the following figures, and for gas path links, the user specified station number appears.



An example of a simple turbojet model is given by the TJET model.

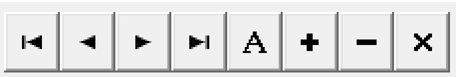


If the new model has been completed, the component data must be entered.

6.4.3.5.3 Entering component data

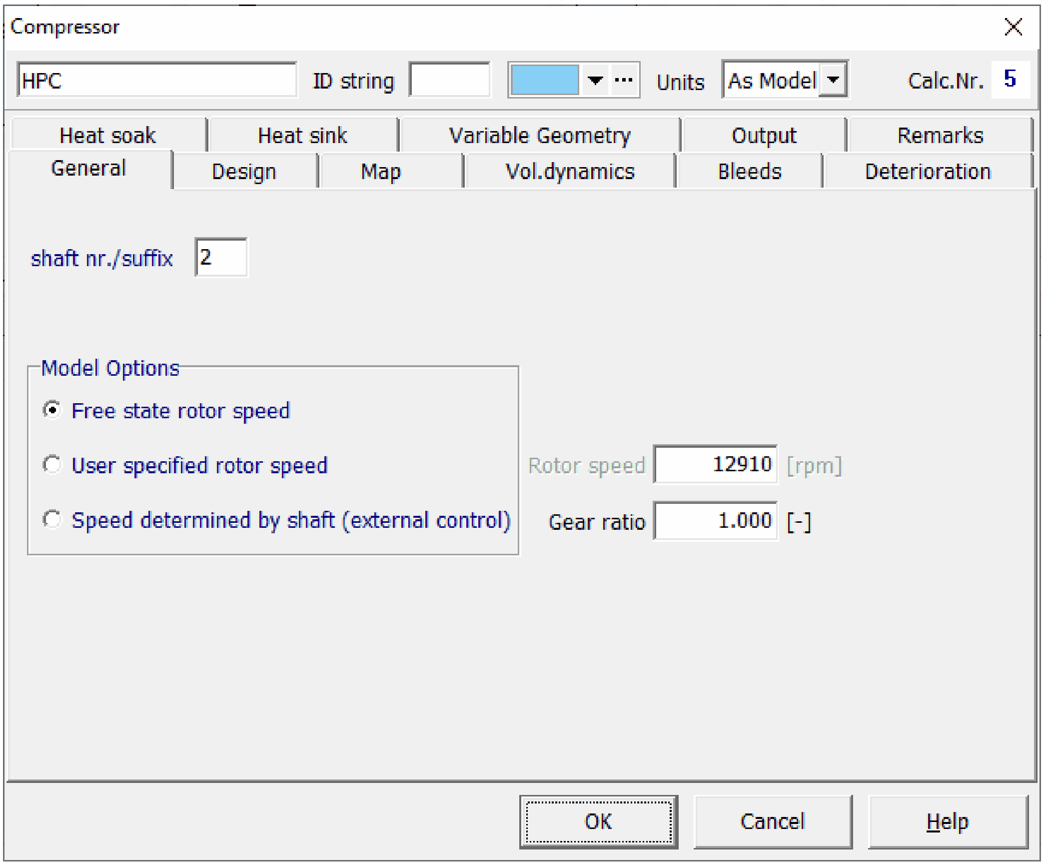
To enter component data, select the desired component icon and double-click it, or right-click and select `Edit` to open the component data window. This data window consists of a header containing some information of the simulation block and a structured collection of tab sheets containing input fields or grids to modify the model properties. The data management functionality decides which input fields or grids are available for editing. Non-editable input fields will be disabled.

The input depends on the type of simulation, model property, configuration, etc. Common input specifiers are e.g. numerical input boxes, text input boxes, radio buttons, check marks, numerical input grids. Numerical input grids are usually controlled by the accompanying grid navigator:



This navigator interface element contains buttons to step through the grid or add and remove rows from the grid. In case such a navigator is not supplied, the 'arrow down' button can be used to add a row to the bottom of the grid, the 'Insert' button can be used to add a row before the current selected row, and the 'Delete' button can be used to delete a row from the grid. Note that a right-click on most grids results in a pop-up menu with some additional options.

The following figure depicts an example of the compressor data form.



The name of the component is located in the top left input field (representing the component ID's in the project XML doc). Changes in component names are propagated into child cases and configurations to ensure data integrity. These component names (representing the component ID's in the project XML doc) can not be adapted anymore in the child nodes; it inherits the name from their parents.

The `ID string` field is used for the identification of the output parameters. When left empty the component number (or `Calc. Nr.`) is used as identifier, e.g. the pressure ratio parameter for the HPC above would be identified by PR_5, with the `ID string` set to "hpc" the output parameter would be named PR_hpc. Note that the name of the component can also be used to identify output parameters, see `Options|Output|Output Identification`. Filling out an ID string is highly recommended since model block re-arrangements (e.g. compressor calculation number becomes 6 after inserting a scheduling block or a case controller) do not result in resetting the output table name array (this way you can more easily compare output between two cases within the same project).

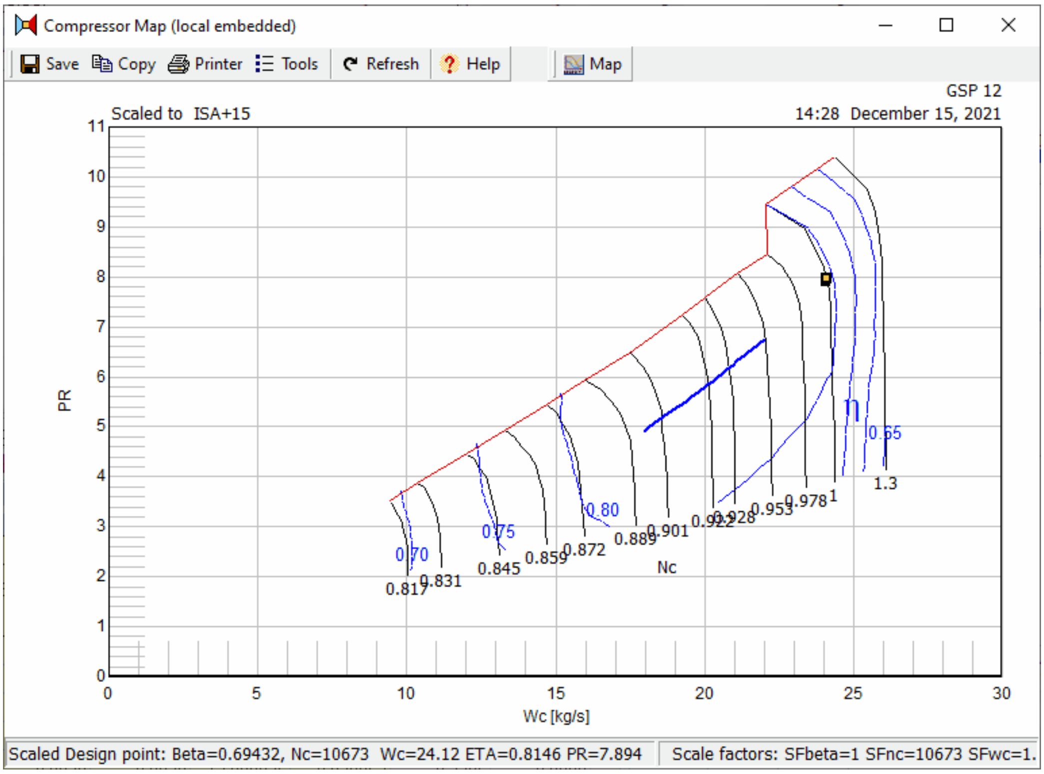
Next to `ID string` input field is the color picker for the data output table. The selected color will be the background of the output parameter grid for the output of this component. A color picker is available where custom colors (preferably light colors) can be selected. Right click for a menu option to revert the color to default/design.

The `Units` drop down box can be used to select the unit system for the component. Default the units are "`As Model`" which can be set in `Output|General|Units` for the complete model. Change the value to "`SI`" or "`Imperial`" to change the unit system from the default for this component only.

The `Calc. Nr.` is the component or calculation number, that is displayed on the Component Icon.

Depending on the component, several tab sheets can be selected, most common of which are:

- `General`
  The `General` tab sheet is used to set general component properties, model options, off-design input parameters, component specific data and component options. After a design point has been calculated, the `General` tab sheet will usually be used to change general component properties for analysis of off-design behavior (constant in time).
- `Design`
  Design values, colored navy blue when using standard Windows colors, are used for specification of the component's design point performance parameters. Use this tab sheet only prior to a new design point calculation. Design values are not used during off-design simulation if corresponding variables are specified in other tab sheets. Note that the design calculation resets off-design variables to their design variables, if `Auto Reset Input to Design` is checked in `Options|General`.
- `Map`
  Many component types have their off-design performance described in *maps*. GSP maps represent non-linear characteristics between 2 to 5 parameters for components such as the fan, and are stored in tabular format in separate ASCII files (or embedded in XML format). The `Map` tab sheet contains controls to select map files, specify map design values and show graphical representations of the map, as displayed in the following figure.

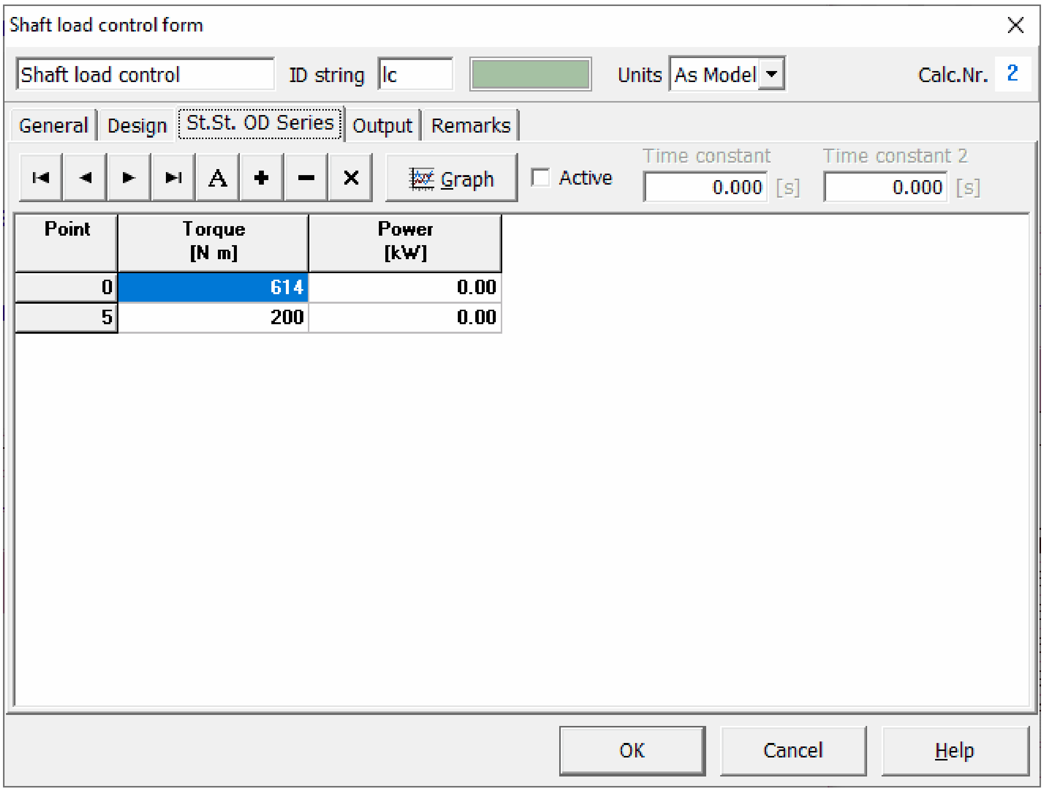


For most maps, parameters corrected for component entry conditions are used. With the map design values, the map can automatically be scaled to the design parameters of the model, e.g. scale a compressor map to a larger compressor (i.e. larger flow rate), a different rotor speed, higher efficiency or different pressure ratio (see section). This allows the use of maps from other gas turbine models with similar components (e.g. a similar axial compressor with about the same number of stages but a lower air flow rate).
Use the `Show Graph` button to show a graphical representation of the map. In section Graphical output it is described how to obtain simulation result operating curves in maps. After design point calculation, the map can be shown with parameters scaled to the

design point. Depending on the type of component, several ways of displaying the relations among the parameters are possible (different parameters at X and Y axes).

- `St. St. Series / Transient`

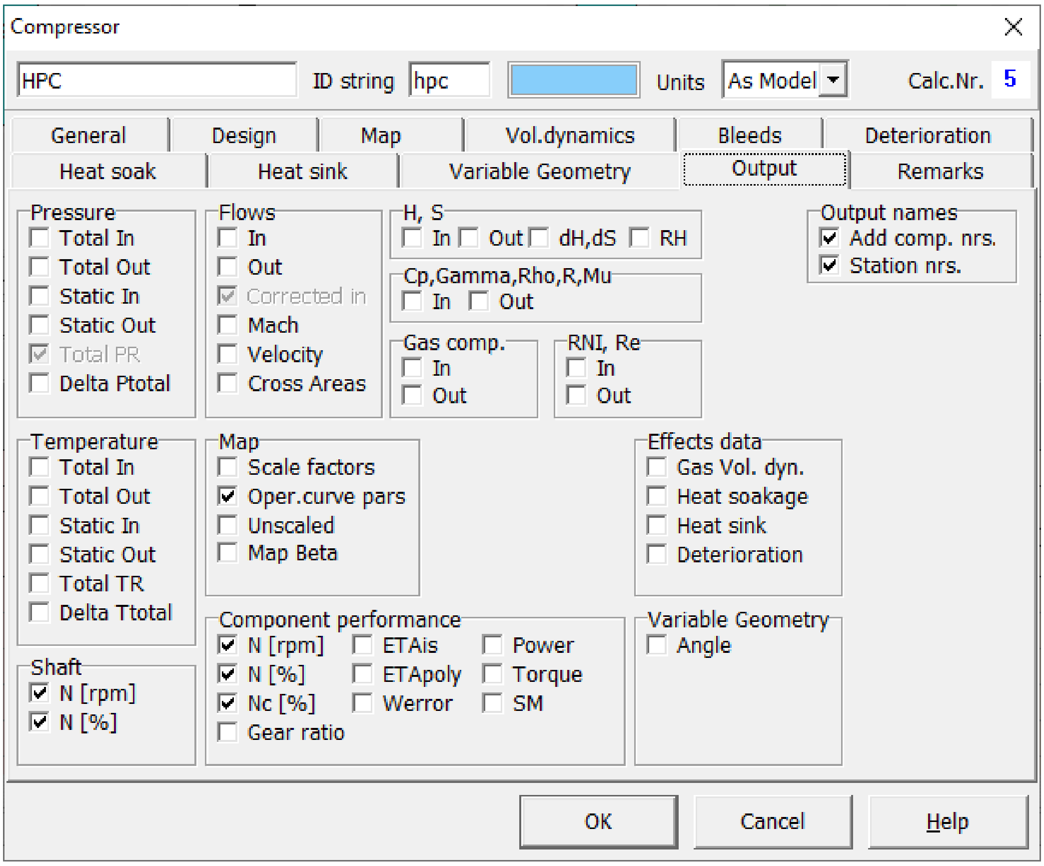
  In the `Transient` tab sheet input parameters can be specified as a function of time in tabular format as displayed. With the `Transient input activated` checkbox the time function can be enabled or disabled during transient or steady state series calculations.
  When disabled or when only one row is present, the single off-design conditions specified in the `General` or other tab sheets are used instead (and input is constant in time). The navigator buttons are used to add, insert and delete rows, while right-clicking on a value in a table and selecting `Edit`, or double-clicking the value allows you to edit the value.



  Use the `Graph` button for a graphical representation of the transient input time functions.

- `Output`

  In the `Output` tab sheet parameters which will be included in the results tables can be specified using check boxes as displayed. You are advised to limit the number of output parameters in order to avoid very wide output tables and large amounts of output data that hinder finding the data of interest.
  In the various output tables and graphs, component numbers and/or station numbers are added to output parameter identifiers for identification where applicable. For example, pressures and temperatures typically apply to engine stations resulting in Pt2 as the total pressure at station 2. Parameters applying to the component are identified with component numbers preceded by the '_' character. The '_' is for distinguishing component from station numbers, for example resulting in PR_2 for the pressure ratio for component number 2. If the `ID string` field at the top of the component data form is filled, this string is used instead to identify the component, for example resulting in `PR_compr` when `compr` is entered in the field.



- `Deterioration`
  With the `Deterioration` tab sheet you can implement deterioration effects for several components. These deterioration effects apply to mass flow, pressure and efficiency.
- `Remarks`
  The `Remarks` tab sheet can be used to record additional descriptions or remarks.

6.4.3.5.4 Calculation order

GSP automatically determines the order of calculation the components in the gas path. The order is indicated by the Calc.Nr. (component number) field and also on top of the component icons in the Model panel, after the model has been initialized. Some elements of the calculation order can be user controlled. This is important for unlinked / non-gas path components such as Additional output parameter, Constant expression and Equation schedulers that depend on other component calculation results. The non-linked components for control are all automatically moved to the start since these are to control model inputs and parameters in expression input fields. The post-processing type components (Additional output parameter etc.) are put at the end since these are using calculated data from the other components.

Ordering unlinked components

Right-clicking the component icon in the model panel, enables the user to move the components forward or backward in the component array. This may be necessary if one component uses data calculation in another similar components. A good example is a series of Additional output parameter components where the next uses the value of the prior additional output parameter.

Inlet calculation order

The inlet calculation order may be important if multiple engines are simulated simultaneously or with multiple inlet engine systems. Inlet calculation order can be controlled by two different mechanisms.

- Use the "Inlet Nr" field in the design tab (the turbojet with the lowest Inlet Nr is calculated first) which takes precedence over the inlet exit station number. If two turbojets have identical "Inlet Nr", then the one with the lowest inlet exit station number is calculated first.

- I no precedence is determined by Inlet nr, you can instead set the inlet exit station number of the turbojet to be calculated first to a value lower than the other turbojet (i.e. if the inlet exit station number of the turbojet to be calculated first is "2", then the inlet exit station number of the other turbojet should be set to a value greater than this (e.g. 2.1, etc.)).

6.4.3.6 Model change status

The left panel of the status bar at the bottom of the project window indicates the modification status of the model. It may either show that the model is `unchanged` or `changed` (e.g. changed component model position, parameters etc.).

6.4.4 Case and configuration management

6.4.4.1 Introduction

In GSP projects, model configurations and cases are managed using a powerful XML data inheritance mechanism. The Reference model represents the single root of the project from which adapted configurations are derived and saved as child nodes in the project tree. For a child node, only the data that deviate from the parent are stored. This means data storage size is minimized, data duplication avoided and loading and saving speed maximized.

When created/added underneath a parent, a child configuration initially is an exact copy of that parent and when saved, no XML data are actually stored. Only after adaptations, either directly by the user or indirectly by GSP as a result of other user actions, XML data identified as deviating from the parent is stored.
The deviating data can be seen in de Details panel view of the project window (the default location is next to the Model panel).
Underneath Configurations, Cases can be added which include input data for actual simulations. Configurations cannot be added under cases.

When this mechanism is understood and projects are carefully built after some analysis of the modeling and simulation objectives in advance, very efficient set-ups can be made. A well organized project inheritance structure offers a very efficient and survey-able modeling environment for a particular tasks. New cases and/or configurations can be easily added as sub-nodes and with minimal data entry efforts, simulations can be run and results analyzed. All model data is stored in the project, so one can always go back to a previous case to analyze results. Moreover, results from different cases can easily be compared in graphs by running cases and adding breaks or group-breaks in between the output table rows.

The following elements are found in the project tree in the project window:

- Project root node with optional project name
- Project Options
- Reference Model
- Configuration
- Case

Note that is is impossible to delete components that have been introduced somewhere in an ancestor. There is a single exception on this rule; case input data components may be deleted from Cases. This is justified by the fact that these components only describe the (operating) input data for the simulations, and have no effect on the configuration of the cycle. Case components which are deleted, are marked inactive; this means that the components are still present in the file, but not loaded or used by the case model.

6.4.4.1.1 XML

The Extensible Markup Language (XML) is a W3C-recommended general-purpose markup language for creating special-purpose markup languages, capable of describing many different kinds of data. In other words: XML is a way of describing data and an XML file can contain the data too, as in a database. It is a simplified subset of Standard Generalized Markup Language (SGML). Its primary purpose is to facilitate the sharing of data across different systems, particularly systems connected via the Internet. Languages based on XML (for example, Geography Markup Language (GML), RDF/XML, RSS, Atom, MathML, XHTML, SVG, and MusicXML) are defined in a formal way, allowing programs to modify and validate documents in these languages without prior knowledge of their form.

6.4.4.2 Configurations

A configuration represents a mode design configuration that can be run (simulated) in a case. In the project tree it is a child node of a Base configuration odel or an other configuration. The modeler is allowed to add gas path components and control components. Basically all components that setup the mathematical model (set of states and error equations) can be inserted here. The Reference Model itself is also a configuration.

In the configuration model the modeler specifies the basic (or enhanced with respect to the Reference Model) configuration of the model of the engine. This permits the modeler to change the model specific configuration data (such as shaft ID's e.g.) and design input field data. Refer to input access rules for detailed rules for input access.

A `Configuration` instance cannot be run.

6.4.4.2.1 Reference Model

The `Base Configuration model` (`BaseCfg` or `Reference model` in GSP 11) is the root configuration from which all Configurations and Case models will inherit. This top configuration usually is the engine configuration from which a design or performance analysis study is started. There is no restriction in the model arrangements in the configurations derived from the reference model. An extreme example is an empty reference model with a number of different sub-configurations. In the future this variant may be replaced by a set-up allowing multiple reference models in a GSP project.

A `Base Configuration` model is similar to the configuration model with respect to access to certain input. In the `Base Configuration` model the modeler specifies the configuration of the model of the engine. This permits the modeler to change the model specific configuration data (such as shaft ID's e.g.) and design input field data. Refer to input access rules for detailed rules for input access.

In this model the modeler is allowed to place gas path components and control components.

A `Base Configuration model` instance cannot be run.

6.4.4.3 Cases

A Case represents an instance of a predefined set of simulation input conditions for a configuration instance. In the project tree, it is an child node of a `Configuration` or another `Case`. In a case, simulation input can be specified in Case components, or other components that include simulation input specifiers such as specific operating conditions. Changes to the model configurations are inhibited when in a Case node in the project tree.

Only `Cases` can be run when an appropriate case type has been set. The case type determines the access to the input data fields, please refer to input access rules for detailed rules. The model in a case is "frozen", i.e. that it is not allowed to alter the component arrangement (moving or rotating the blocks) of the inherited components.

When a case is selected in the project tree and properly configured, the simulation can be started from the project toolbar menu.

6.4.4.3.1 Case types

Initially when the simulation case is created the default case type is `Select mode...` and is activated for the modeler to instantly change to a valid simulation case type. The simulation case type can be set in the second column of the Project tree panel window. The "Run button" automatically disables when no case or case type is selected. Selecting the second column or creating a new case configuration, a selection box will pop-up. The following simulation modes can be selected:

- `Design`
  Perform a design point calculation.
- `Design Series`
  Perform a series of design point calculations. Using this simulation case type requires a case components library component supporting DP series calculation.
- `Steady-State`
  Perform a single off-design steady-state calculation.
- `St.St. Series`
  Perform a series of off-design steady-state calculations. This requires the use of a case component or enabled transient input data.
- `Transient`
  Perform a transient calculation.

Note that the case type is part of the project data, not part of the model data for which changes require action to save to XML. Case type is directly saved to XML from the GUI: it is therefore seen on the left in the project tree. Only when a change in case type has required a model data change (such as when changing to non-series type and series data had been deleted) you will be prompted to save the case.

6.4.4.4 Input access rules

The project based modeling approach with the structured models for configurations (and base configuration model) and cases provides the user with different models to setup different input of the model. This requires a consistent set of rules for input field access. Summarized these rules are:

1. 'Configuration-only' and 'Case-only' fields:
   Only enabled when either in configuration (or base configuration/reference model) or case project level.
2. DP and OD input field access rules implemented:
   Navy blue or otherwise 'DP tagged' input fields only enabled in either configuration project level or Design point or design series case type.
3. Transient/Series or single point:
   With single point case type, control component Transient/Series tabs are invisible.

Optional extra access (controlled by `Case Management` options of the environment options):
(note that this is optional, for some specific installations this has been disabled on user request, for questions please contact NLR)

4. For rule 1., allow case input in configurations (note that vice versa (config input in cases) is **NEVER** allowed).
5. Allow OD input in DP cases (note that vice versa, DP input in OD, is **NEVER** allowed).
6. Allow editing components with no case type selected.

6.4.4.5 Copying and pasting configuration and case nodes

When copying and the pasting a configuration or case node in the project tree, only the changes of a case or config relative to the parent pasted, resulting in a new node inheriting all from the new parent except the changes in the original relative to the original parent. When components do not exist in the new parent, the
component is created and adapted with the original change data only; the rest is remaining on the initial component data defaults (as there is nothing to inherit from).

Note that only with the `Allow copy-paste nodes outside parent subtree` Environment Case management option active, nodes can be copied outside the parent. This is to prevent messed up projects when the user does not fully understand how the project data are handled inside.

6.5 Running Simulations

Running simulations and analyzing results are the eventual objectives of building models. After model configurations have been specified and cases and case types defined using the configuration and case management functions, simulation input can be specified in the run case selected.
After starting the simulation the convergence progress can be monitored with simulation progress box or, for more details, the convergence monitor.

The model calculation status indicates the status of the simulation.

A simulation always requires the model to be initialized. Next, a design point simulation is required to establish the reference point for any off-design simulations. The user can re-initialize the model requiring a new Design point calculation before any off-design simulations can be started again.
Multiple Design points can also be calculated using the Design point series case type.
Off-design simulations may be

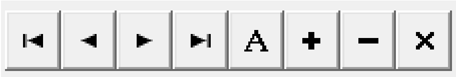
- single point steady state calculations,
- steady-state series calculations ('OD sweeps'),
- or transient simulations.

6.5.1 Specifying simulation input

Simulation input normally are all data that define the gas turbine engine operating conditions in terms of ambient conditions, power setting, engine control settings, engine component condition (deterioration), loads etc. These data should all be defined in the cases in the project tree from which the simulations are run. Except for the ambient or flight conditions specified in the separate Ambient/Flight conditions window, all simulation input data are specified in GSP components depending on the Input access rules. Specific components for simulation input include control components, case controllers, schedulers and auxiliary components.
For both steady-state series and transient simulations, transient input is required to specify the input as a function of point nr. (steady-states) or time (transient).

6.5.1.1 Transient input

For transient simulations, input parameters need to be specified as a function of time. The engine's response to that input change is then calculated. In GSP the input-time functions are specified using time-input tables in the `Transient` tab sheet specification windows.

Edit the time-parameter table using the navigator buttons  to go to the first line, previous line, next line or last line or to add, insert or delete a line in the table.To edit time of an existing row, right-click with the mouse on the row and select `Edit`. Use the `Transient input activated` checkbox to enable or disable the time function during transient or steady state series calculations. When disabled or when only one row is present, the off-design conditions are used instead (and input is constant in time).
If more than one input transient is specified, transient inputs work simultaneously during simulation and the shortest time range determines the simulation end time.

Transient input can be made visible graphically by clicking the `Graph` button.

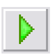
Note that changes in infinite short times are invalid: a gas turbine engine control system would never be able to realize it and GSP will likely not be able to simulate it. So make sure you enter time values with reasonable (not too small) positive increments between the rows.

6.5.2 Starting a simulation

After having built or opened a GSP model and having specified simulation input data parameters and maps, simulations can be run. Before simulation, the model configuration and component data are checked. If the model is invalid, the calculation is canceled and an error is reported. If the model is valid, five types of simulation calculations can be run, depending on the case type.

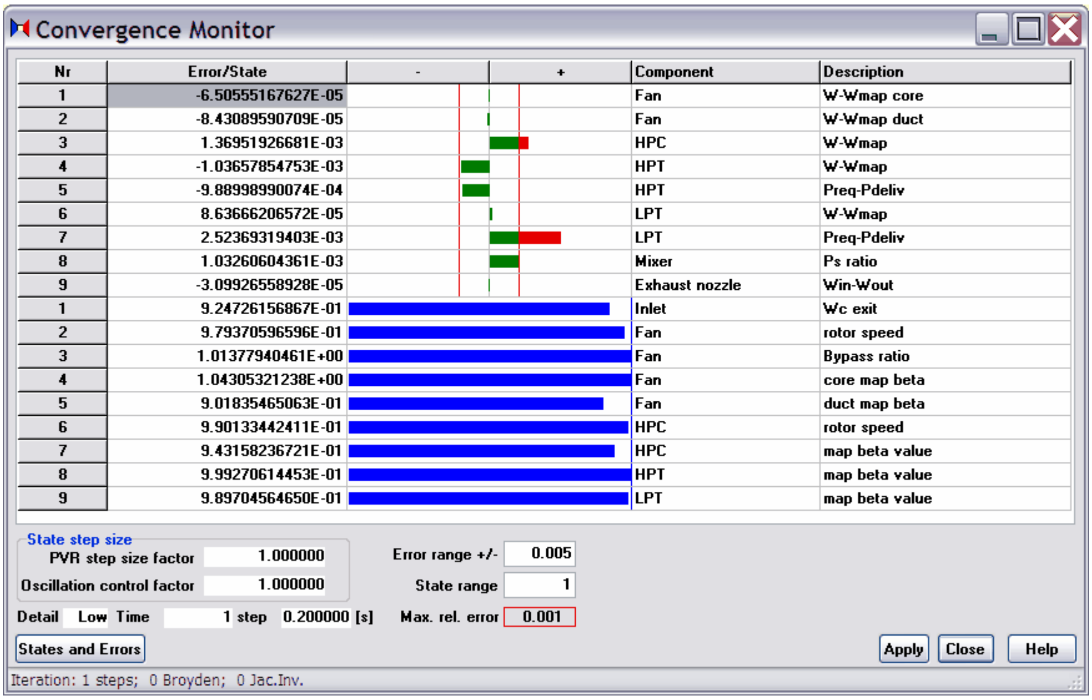
- Design point
- Design point series
- Steady state point
- Steady state series
- Transient

Before starting a simulation, it is advised to first consider which parameters are of interest for output and specify the output parameters in the `Output` tab sheets of the component editing forms.

Use the  ('play') button (shortcut key F9) to start the simulation from the project toolbar. To start the simulation a Case must be selected in the project tree in order to enable the button (when disabled it looks like ).

6.5.3 Convergence monitor

This window graphically displays the error convergence during simulations. This is useful to analyze simulation stability and convergence problems.



The windows displays both the error variable (top entries in the grid) and the relative state variable (bottom entries in the grid, blue colored bars). Convergence is obtained after all error variables have errors that are smaller than the accuracy defined in the Iteration options (displayed in green if the error is smaller than the defined accuracy). When the error bar has a red component the error has not converged to the accuracy. Use the `Error range +/-` to adjust the scale of the error bars if no green bar is displayed, or displayed too small. Use the `state range` to define the scale of the state variable (blue) bars.

6.5.4 Model calculation status

The middle panel on the model window status bar indicates the status of the calculation process:

- `Not initialized`
  Indicates a model just loaded or modified
- `Reset / Not initialized`
  Indicates a model being reset
- `Input changed`
  Input parameters changed since last calculation
- `Initialized`
  The model has gone through initialization calculations
- `Checked`
  The model integrity has been checked and is ready for simulation
- `Design point`
  A design point calculation has been succeeded
- `Steady State`
  A steady state point calculation has been succeeded

- `Iteration progress information`
  Indicates status during and at the end of iterations

6.5.5 Init Button

Click the (Init) button of the project toolbar to (re-)initialize the model. Upon pressing this button several checks/initializations on the model are performed, e.g.:

- check whether all components are connected,
- create component calculation list,
- component label numbers are refreshed,
- checking of duplicate station ID's,
- checking of duplicate output parameters (e.g. shaft speed output),
- defines component configuration, States and Error vars,
- executes init procedure of components (e.g. creates a list of output parameters to be selected from a drop down list),
- check bleeds,
- check whether new tables have to be created,
- resets all global output parameters.

You may want to Reinitialize the model after a simulation prior to a next one in order to have the model state reside at its optimal starting point for the iteration. For example: after ending a simulation at near IDLE power, much deviating inlet conditions, deterioration or other severe operating condition deviations, starting the simulation with a different input may require GSP to iterate to instable/impossible/conflicting operating regions. After a decel to IDLE for example, a successive simulation at full power will cause over-rich mixture in the combustor, resulting in failure to calculated realistic combustor exit temperatures.

6.5.6 Design point calculation

After having specified a `Design` case type, run a design point calculation by clicking the run button (green arrow) or press F9 in the project window. With the design point calculation, the gas turbine design point performance is fixed to represent a particular gas turbine configuration. The components are "sized" to the design point using data from the `Design` tab sheets (data from other tab sheets that do normally not change in off-design operation other than by user specification are also used for the design calculation such as compressor bleed and turbine cooling data). The turbine design point calculation for example determines turbine flow rate capacity that represents geometry in terms of effective flow area of the stator nozzle. For the design point calculation, the component maps are not used but scale factors are calculated instead for scaling of the maps during subsequent off-design calculations (see the following sections). A design point calculation is always necessary before off-design, steady state or transient calculations, since the design point is used as the reference point for off-design operating points. Note that a design point calculation resets off-design input parameters in the General and certain other tab sheets back to their design values, if `Auto Reset Input to Design` is checked in `Options|General`.

6.5.6.1 Run DP and reset single point OD input to DP values

Click the button of the project toolbar to run a design point calculation and afterwards reset the single point off-design input to the design point input values. This will reset the off-design model to design input, resulting in identical design/off-design output values.

6.5.7 Design point series calculation

A series of Design points ('DP sweep') can be calculated for design point studies only in conjunction with specific case input components from the Case Components Library. DP sweeps are typically used for engine design point performance analysis using carpet plot output.

6.5.8 Off-design calculations

After the design point has been calculated, both steady state and transient off-design simulation calculations can be performed. Off-design operating conditions can be specified in many ways such as:

- Changing Flight conditions
- Fuel Flow or control-system power setting
- Compressor bleed variation
- Turbine power or torque variation, using the Power Turbine Load control

Off-design conditions may also be specified in the component data windows (usually, the basic off-design parameters are located in the `General` tab sheet).

The basic model layout and the design point should not be changed (i.e. design point data, changes which affect the number of states or component interconnections, numbers of bleed- or cooling flows etc.). In the latter case, GSP will generate an error message automatically or reset the model so as to require a new design point calculation first.
Changing component deterioration and variable geometry effects may also be calculated as off-design calculations.

6.5.9 Steady State calculations

A steady state point corresponds with a fully stabilized condition of the gas turbine at a specific off-design condition. Main user input affecting off-design operation is specified in the `General` tab sheet. Several other tab sheet are available for more specific deviations from the design point operating conditions such as bleed flows, pressure losses, deterioration, variable geometry, heat soakage and volume effects etc., depending on the component type. Run a steady state point calculation by clicking the `Steady State` button in the model window. Note that a design point calculation resets off-design input parameters in the `General` and certain other tab sheets back to their design values, if `Auto Reset Input to Design` is checked in `Options|General`.

6.5.10 Steady State series calculations

The steady state series calculation mode enables the calculation of a series of steady state operating points with varying off-design operating conditions. These "parameter sweeps" are applied to determine the relation between engine performance and a specific parameter, for example fuel flow from idle to max power. Therefore, steady state operating point calculations are required over a wide operating range. Usually only a single input parameter is varied for a parameter sweep to obtain a clear indication of the effect of the parameter, but multiple parameters may be varied simultaneously if desired. GSP steady state series calculations offer a flexible way to apply input parameter sweeps using the time-function tables also used for transient simulation input. The tabular time input function is used to specify the input parameter values at which a steady state calculation is required. Although the time parameter has no physical meaning for a steady state series, it is used as a scale to define a series of input parameter values. The transient output time step setting determines the intermediate

input values for steady state calculations between the specified input values in the transient input table.
Start a steady state series calculation by clicking `St.St.Series` in the model window.

6.5.11 Transient calculations

Transient simulations represent engine responses to variations in time of one or more operating conditions. Transient component performance is calculated using the quasi steady state approach with steady state component maps. The transient effects included in GSP are rotor inertia, volume and heat soakage effects and, if control system models are added, control system dynamics. Rotor inertia, directly affecting rotor speed acceleration rates, usually is the dominating factor for transient performance. Volume and heat soakage effects may be calculated for each gas path component but often are relatively small and therefore disabled. They only significantly affect performance in special cases such as recuperated cycles.
For GSP transient simulations, the input-time functions are specified using time-input tables in the `Transient` tab sheets in applicable component windows and in the `Ambient conditions` window. Run a transient series calculation by clicking `Transient` in the model window. For a transient calculation at least one component must have a transient input specified for a variable. This transient input and the time range define the simulating time range. If more than one input transient is specified, these inputs work simultaneously during simulation and the shortest time range determines the simulation end time. If successful, a result table showing the transient series data will appear and data can be presented graphically. Transient input tables may be deactivated using the `Transient input activated` check box. Click the `Graph` button to see a graphical presentation of the transient input table.

6.6 Simulation Results

Simulations generate output data/results. In GSP there are several ways to display the results.

- Results are output to tables. Generally, tabular data does not provide a very clear representation of the results. Therefore, GSP provides a simple method of graphically visualizing relations among parameters from the output tables.
- A multi purpose Report Editor is available to output results e.g. GSP can output individual operating point data to a textual report in this Report editor.

6.6.1 Output tables

Simulation results can be output to tables with columns for the user-selected output parameters and each row representing an operating point. SQL based database storage is used which can hold up to 256 columns and any number of rows. Per case model, a separate output table is stored. The table can contain operating point results from design point, steady state, steady state series and transient calculations. Note that output data tables from different cases can be compared by selecting the check boxes in from of the case name of the project tree panel.

The output table contains default columns which are supplemented by the selected output parameters of the Ambient conditions, Model components, System performance and calculated parameters output parameters. The default columns are `Point/time`, `Type`, `curve #`, `group #` and `Comment` columns.

	General				
	Point / time	Type	curve #	group #	Comment
▶	0	Off-design	0	0	
	0.2	Off-design	0	0	
	0.4	Off-design	0	0	
	0.6	Off-design	0	0	

After each operating point calculation, rows are automatically added (with or without confirmation) depending on the settings in `Options|Output` in the main window. An example of steady state tabular output is displayed in the following figure.

Output table

Drag a column header here to group by that column

	Prim.Combustor									HPT					
om	FARexit_cc [-]	FARstoich_cc [-]	Rhofuel_cc [kg/m³]	Hfueltank_cc [kJ/kg]	Hfuelin_cc [kJ/kg]	ETAsp_cc [-]	ETA_cc [-]	Delta3 [-]	dPtrelaerodyn_cc [-]	Wc4 [kg/s]	PR_hpt [-]	N_hpt [rpm]	N%_hpt [%]	Nc_hpt [%]	Wcompc_hpt [kg/s]
.55	0.016781	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	10.329484	0.044958	5.5615	3.7891	11189	79.92	98.64	5.560
.50	0.016376	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	9.949787	0.045154	5.5606	3.7928	11111	79.36	98.69	5.5598
.46	0.015967	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	9.585434	0.045382	5.5602	3.7967	11037	78.83	98.78	5.5594
.42	0.015572	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	9.222392	0.045584	5.5596	3.8014	10960	78.28	98.86	5.559
.38	0.015183	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	8.865677	0.045786	5.5593	3.8063	10884	77.74	98.94	5.5587
.34	0.014801	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	8.511066	0.045982	5.5588	3.8101	10804	77.17	99.01	5.5584
.31	0.01442	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	8.169395	0.046193	5.5585	3.8136	10726	76.61	99.09	5.558
.28	0.014031	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	7.845473	0.046431	5.5576	3.8163	10650	76.07	99.21	5.5575
.26	0.013612	0.06817	800.0000	-1446.959	-1446.959	0.9940	0.9940	7.562004	0.046758	5.5553	3.818	10584	75.60	99.42	5.5566
.23	0.013203	0.06817	800.0000	-1446.959	-1446.959	0.9938	0.9938	7.279008	0.047135	5.5567	3.8182	10517	75.12	99.64	5.5557
.21	0.012797	0.06817	800.0000	-1446.959	-1446.959	0.9936	0.9936	6.999562	0.047479	5.5548	3.8171	10455	74.68	99.92	5.5546
.19	0.012412	0.06817	800.0000	-1446.959	-1446.959	0.9934	0.9934	6.691713	0.047789	5.554	3.813	10384	74.17	100.12	5.5536
.17	0.012065	0.06817	800.0000	-1446.959	-1446.959	0.9932	0.9932	6.349872	0.047927	5.5486	3.8005	10298	73.56	100.20	5.5533
.15	0.011745	0.06817	800.0000	-1446.959	-1446.959	0.9930	0.9930	5.999604	0.048147	5.5533	3.7875	10205	72.89	100.21	5.5532
.13	0.01142	0.06817	800.0000	-1446.959	-1446.959	0.9930	0.9930	5.678137	0.04831	5.5528	3.7664	10111	72.22	100.23	5.5532
.11	0.0111	0.06817	800.0000	-1446.959	-1446.959	0.9930	0.9930	5.36849	0.048476	5.5526	3.7452	10017	71.55	100.25	5.5531

	LPT						Duct	
om	Wc4.5 [kg/s]	PR_lpt [-]	N_lpt [rpm]	N%_lpt [%]	Nc_lpt [%]	Wcompc_lpt [kg/s]	Wc1.3 [kg/s]	dPtrelaero_d [-]
.55	21.4399	1.8633	7080	70.24	79.69	21.4416	14.413	0.121559
.50	21.4554	1.8277	6910	68.56	78.38	21.4564	14.1985	0.117968
.46	21.4703	1.7916	6746	66.93	77.12	21.4713	13.9571	0.11399
.42	21.4891	1.7526	6551	64.99	75.49	21.4899	13.7105	0.109997
.38	21.5081	1.7118	6348	62.98	73.74	21.5089	13.4536	0.105914
.34	21.5229	1.6695	6146	60.97	71.99	21.5233	13.1708	0.101508
.31	21.5348	1.6268	5948	59.01	70.25	21.5352	12.868	0.096894
.28	21.5434	1.585	5751	57.05	68.50	21.5433	12.5296	0.091865
.26	21.5491	1.5494	5538	54.95	66.54	21.5459	12.0451	0.084897
.23	21.5393	1.5115	5322	52.79	64.49	21.5407	11.5175	0.077623
.21	21.5279	1.4739	5113	50.72	62.51	21.5271	10.9582	0.070267
.19	21.5031	1.4306	4881	48.42	60.20	21.501	10.2729	0.061753
.17	21.4423	1.3816	4664	46.27	58.05	21.4473	9.6767	0.054794
.15	21.3698	1.3291	4444	44.08	55.81	21.3698	9.0582	0.048013
.13	21.2609	1.281	4250	42.16	53.87	21.2604	8.4326	0.04161
.11	21.1514	1.2325	4060	40.28	51.94	21.149	7.7786	0.035406

30 of 30

30 records | 5 dataseries / curves

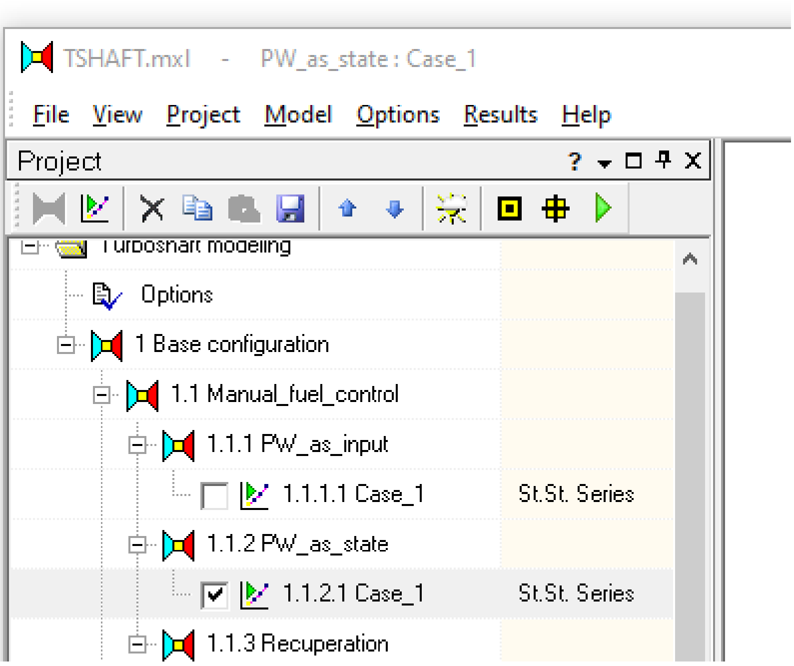
If you specify additional output parameters in the `Output` tab sheet of components while tables already exist, GSP must redefine a new output table with the new columns added. The `Save/New tables` menu item, used for saving existing steady state and transient data and/or defining new tables, pops-up automatically.

Tablename / ID is shown on the statusbar of the table window, in the lower right corner in case the user want to look into the database manually with e.g. FlameRobin FireBird database management tool for offline usage.

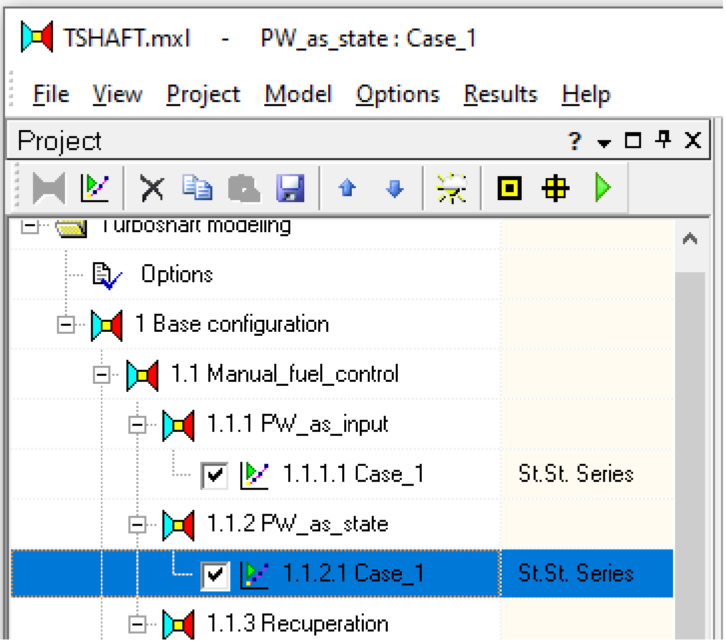
6.6.1.1 Database storage

As of version GSP 12, a sophisticated SQL database storage system is introduced. This new system allows to combine output results of various case models, the data in common will be displayed in a table view so that it is easier to compare results from various case runs. To create a combined view, the check box before other run cases than the currently selected need to be marked/selected. From the combined table view the modeler will be able to graph the results.

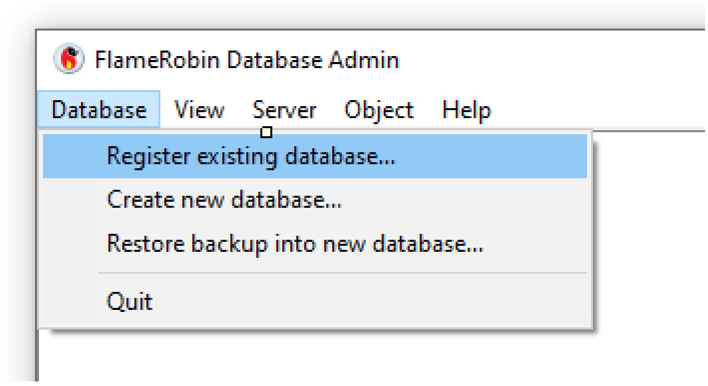
The following will show the run case results of the selected case (case 1.1.2.1):



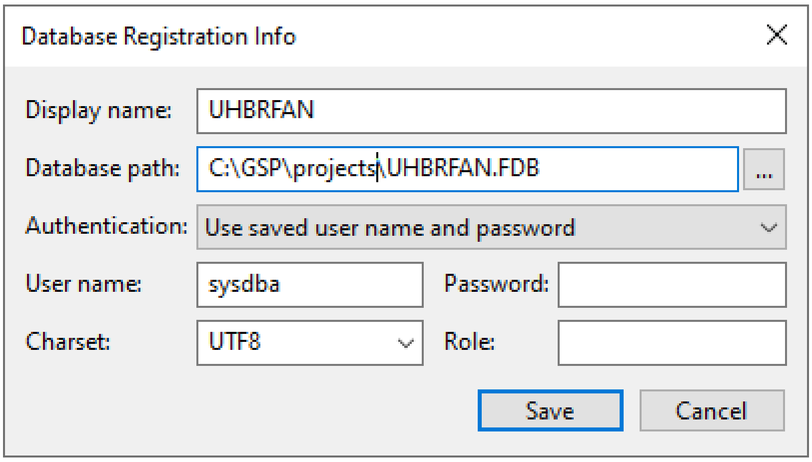
The following will add (Union) data from the non-focused but checked run case (case 1.1.1.1):



The database storage system is the open source Firebird database system. This is an SQL based storage system that creates .FDB storage file extensions. These databases can be opened by tools such as FlameRobin. When having FlameRobin installed, existing database files can be added for viewing/editing:



Make sure the following input is used (the UHBRFAN model is just an example, use your own database file):



6.6.1.2 Table navigation

Use the set of buttons to navigate through and do operations on the dataset

The embedded navigator displays built-in buttons that enable a user to scroll forward or backward through records one at a time, go to the first record, go to the last record, go to the last record in view (page), go to the first record in view (page), insert a new record, post data changes, cancel data changes,delete a record, set a bookmark, go to bookmark and add filter. It also features a text string that specifies the current record and the total record count.

6.6.1.3 Table options

The table toolbar options (from left to right, the "pipe" or separator is depicted below in the overview by a blank line between items):

- `Show graph`
  Display graph window for this table, displaying multiple graphs for one or more simulation sessions

- `Insert a break before the selected row`
  The function of a break is to separate sets of points. Datasets separated by breaks allow multiple graphs for one or more simulation sessions to be displayed with separate line styles and colors specified in the graphs options.
- `Append a break after the last row`

- `Insert a group break before the selected row`
  The function of the group break is to separate several sets of data points to identify surface or carpet data sets
- `Append a group break after the last row`

- `Copy to clipboard`
  Copy the selected table data to the clipboard.
- `Copy all to clipboard`
  Copy all the table data to the clipboard.
- `Save table to file`
  Export the result table as a CSV File. Press `Shift` key to export selection only.
- `Save table to Excel file`
  Export the result table as an Excel File. Press `Shift` key to export selection only. This exports the table output with colors and headers (WYSIWYG).

- `Clear table`
  Add a previously stored database file to the current set, note that the parameter set must be equal. Default/initial path equals the model file path.
- `Reset table layout to original`
  Reset the table layout to the original layout; e.g. use this when parameters (columns) are removed from the view to make them re-appear

- `Show number of curves`
  A dialog will be presented outputting the total number of datasets in the data table.
- `Renumber the curves in the output table`
  Renumbering of the curves and groups in the (joined) output table

- `Help`
  Open the help file.

6.6.1.4 Calculated columns

New columns composed as functions of existing table columns can be created or edited by:

- Right-clicking the appropriate column and selecting `Add Calc Column` from the pop-up menu in the steady state or Transient output table.
- Adding/editing expressions in the model Output Options (go to the model windows `Options`|`Output`|tabsheet `Calculated Expressions`)

This feature is convenient to convert parameters to other units (just multiply with a factor), derive corrected parameters (divide by temperature ratio or pressure ratio) or introduce new variables based on two existing variables (for example: add Engine Pressure Ratio (EPR) from exhaust pressure and inlet pressure). Use the `Select to insert column name` Combo box to add field names' in the expression and add Comments or Unit names if desired in the Comment/Unit field. Expressions can be constructed using the Parser and the drop down box containing all the output parameters selected on the output tabsheets of the components. Note that if selected output parameters names containing a dot (.) will be modified by a dollar ($) sign. The numeric format can be changed to user specifications.

For advanced use map or tables can be used to scale the calculated parameter to a certain schedule or map. When a table (simple schedule) is used only fill out the Xmap_in parameter, for using a map both Xmap_in- and Ymap_in parameter should be filled out.

The expressions are saved with the model and are automatically recalculated after re-opening a model and running simulations. If the expression cannot be calculated (for example because output fields have been deleted or definitions/component nrs. changed), the status field indicates it as **invalid** and the expression requires adaptation. Otherwise **OK** indicates a successfully evaluated expression.

6.6.1.4.1 Format

The format string is constructed by using a 0 (zero) for compulsory digits, and a # (hash) for optional digits.

The following table shows some sample formats and the results produced when the formats are applied to different values:

Format string	1234	-1234	0.5	0	0.531
(none)	1234	-1234	0.5	0	0.531
0	1234	-1234	1	0	1
0.#	1234	-1234	0.5	0	0.5
0.0#	1234.0	-1234.0	0.5	0.0	0.53
0.0##	1234.0	-1234.0	0.5	0.0	0.531
0.00	1234.00	-1234.00	0.50	0.00	0.53
#.##	1234	-1234	.5		0.53
#,##0.00	1,234.00	-1,234.00	0.50	0.00	0.53
#,##0.00;(#,##0.00)	1,234.00	(1,234.00)	0.50	0.00	0.53
#,##0.00;;Zero	1,234.00	-1,234.00	0.50	Zero	0.53
0.000E+00	1.234E+3	-1.234E+3	5.000E-01	0.000E+00	5.310E-01
#.###E-0	1.234E3	-1.234E3	5E-1	0E0	5.31E-1

6.6.1.4.2 Equation Parser

The equation parser provides functions to parse and evaluate text strings into math equations. The text string can be constructed manually or using the buttons to insert operators (+, -, *, /, ^), parenthesis and GSP output parameters (specified on the components output property tab sheet) using the combobox.

Besides the normal math operators, the following functions can be used (note that the arguments in the function can be functions as well):

- `ABS(arg1),`
  returns the absolute value of arg1
- `Exp(arg1),`
  exponent using the base natural logarithm value (e = 2.7182818...)
- `Heav(arg1),`
  step function, returning 0 when arg1 < 0 else 1
- `Ln(arg1),`
  natural logarithm of arg1
- `Max(arg1, arg2),`
  returns arg2 if arg1 < arg2 else arg1
- `Min(arg1, arg2),`
  returns arg1 if arg1 < arg2 else arg2

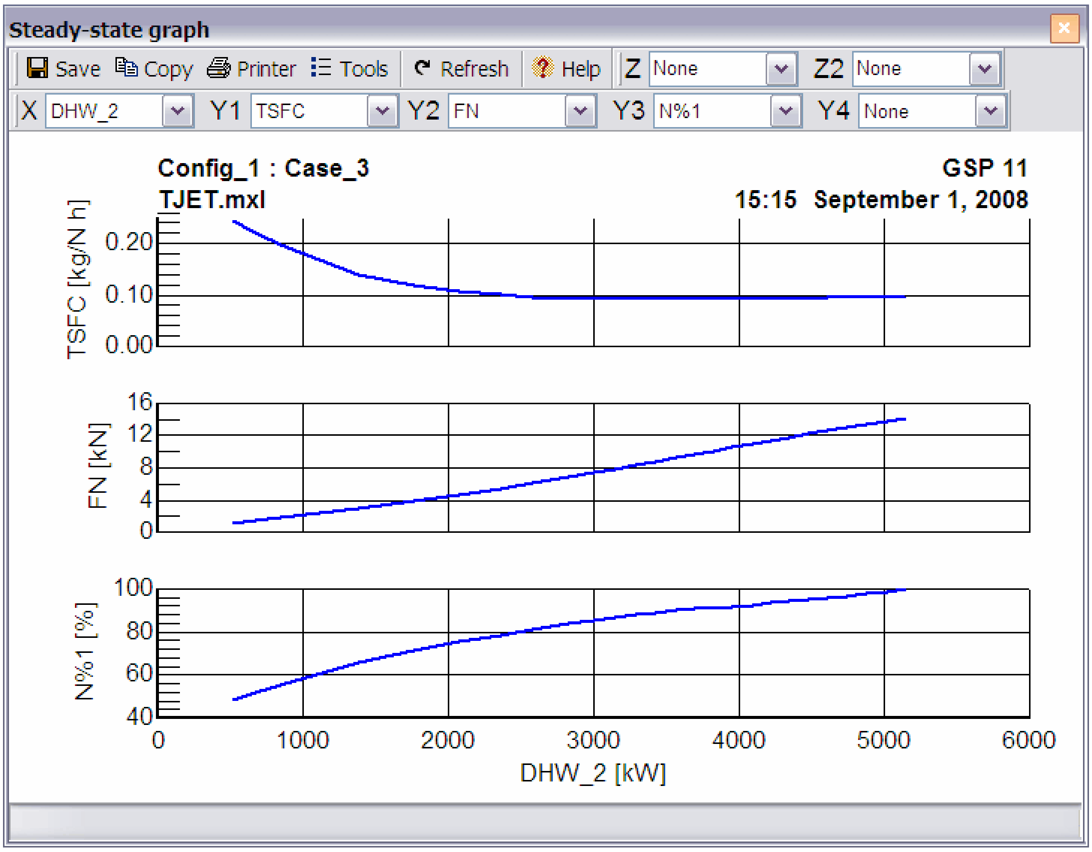
- `Sqrt(arg1),`
  returns the square root of arg1
- `Sign(arg1),`
  returns -1 if arg1 < 0, 1 if arg1 > 0 or 0 if arg1 = 0
- `Zero(arg1),`
  returns 0 if arg1 = 0, else 1

Note that the dot (.) usually is reserved as decimal separator (depending on computer system settings), and therefore cannot be used in the parser. In GSP the decimal separator is fixed to the dot (.) character. Variable names used in parser expression must therefore not contain the dot (.) when entering the parser string manually. Selecting parameters from a drop-down list (usually provided in the graphical user interface) will substitute the dot (.) with the dollar ($) character.

6.6.2 Graphical output

Graphical representations of the table output as shown in the figure are obtained by clicking the Show graph button in the table output window.

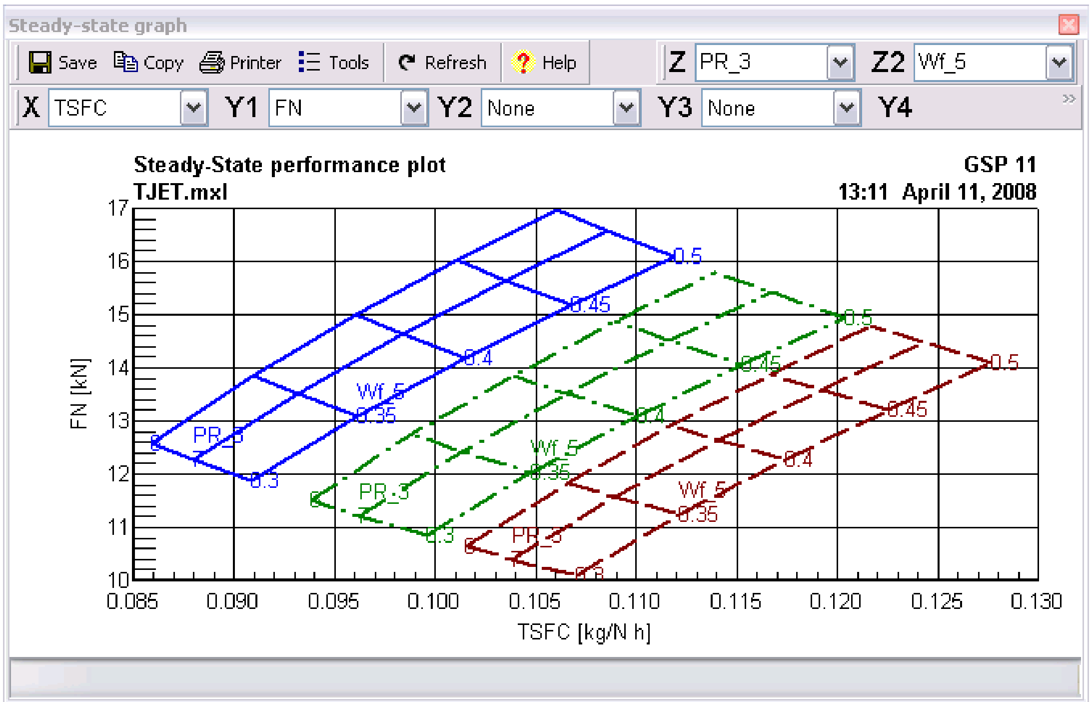
Parameters for the X-axis and up to 4 Y-axes can be specified from those selected for the output table.



Axis and graph options can be changed by selecting `Tools|Options|Axis/Layout`. Multiple simulation results can be placed in one graph if you have inserted breaks between simulation runs. You can show all (up to 4) parameters in a single graph with a common X-axis parameter by selecting `Tools|Options|All curves in single graph`. Graphical layout can be saved to create a graph template for later use or quick switching.

Graphs can be saved or copied to the clipboard or to files in bitmap or meta file format using the file menu. For best results, it is recommended to use the meta file format for export to other application such as word processors.

When options `Tools|Carpet` is enabled the current data will be transformed into a carpet (surface) plot. Parameter labels and values can be displayed using `Tools|Options|Legend|Parameter "Z"` and `Tools|Options|Legend|Parameter "Z2"`, as well as their position on the srceen. Multiple carpets in a single graph can be realized by adding 'group breaks' added in output table to separate different carpets of groups of curves. With every new group, graph settings, line colors etc. start from first again.



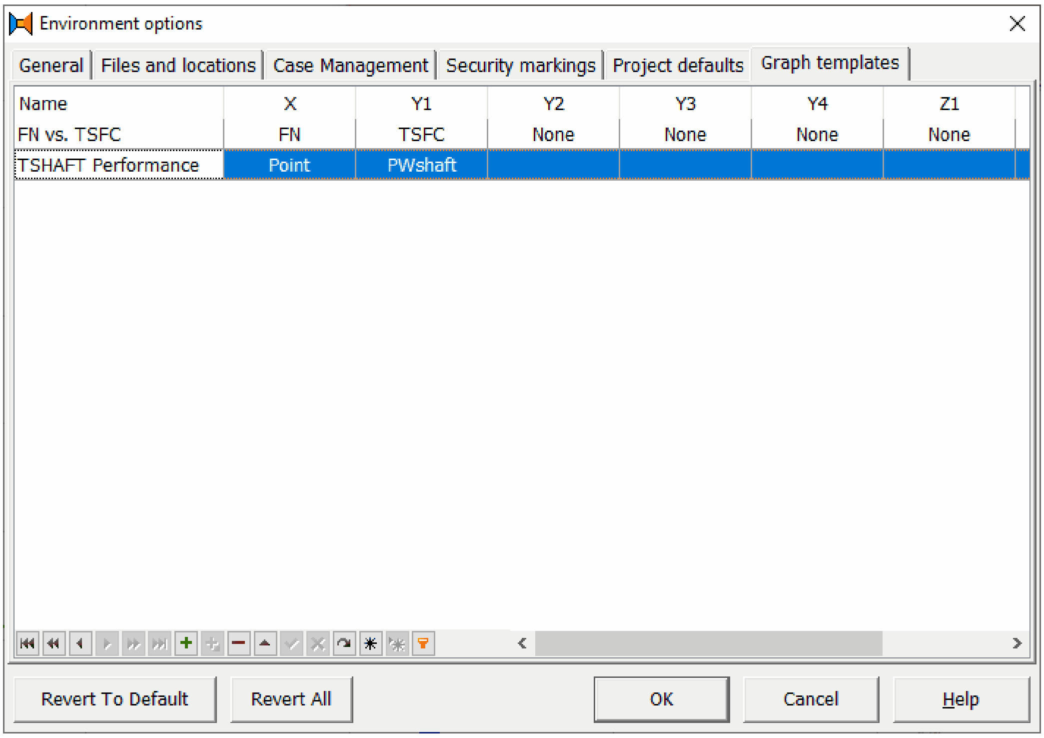
Both steady state and transient simulation results can be plotted in most component maps, as displayed in the next figure, to assess component performance during simulation (e.g. to assess compressor stall margin during acceleration). Before you can view the simulation results, the map needs to be scaled to the model design point using `Tools|Options|Scale to Model des. pnt.` in the map graph window. To show the operating curves, check the `Draw St.St. Points` or `Draw Transient` items in the `Options` menu. The curve line styles can be adapted using the `Line Styles` button, accessible through `Axes/Layout` in the `Options` menu. Note that these line styles are set the same as the steady state respectively transient table graph output line styles.

6.6.2.1 Graph templates

To facilitate quick graphical representation of your tabulated data, templates can be created. The graph template functionality consists of 2 elements:

- Selecting a stored template or adding a new template (the current graphical view)
  Use the toolbar:

to save a graphical view of tabulated data (press the button with the '+' sign, or alternatively press `Tools|Store graph as template`) of the `Steady-State graph` or `Transient graph` tab sheet.
- Storage in the Environment options:

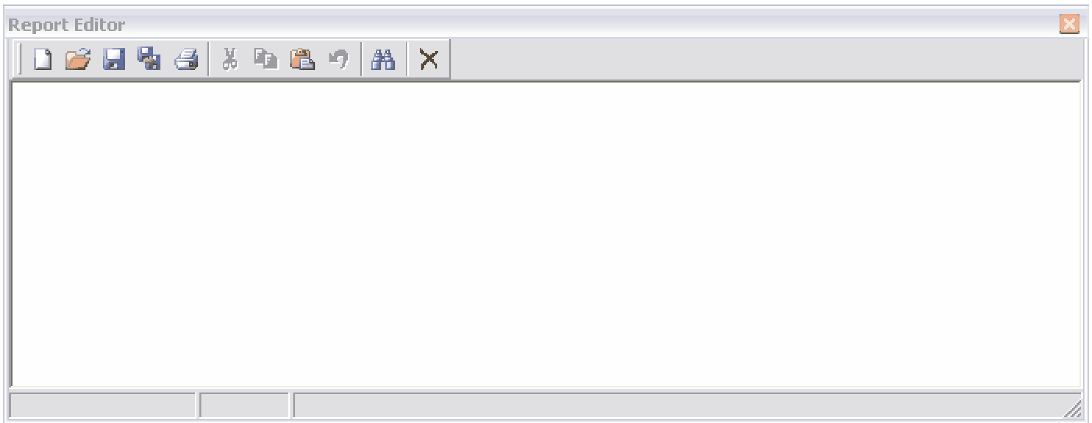


6.6.3 Report Editor

The Report editor is used to display various types of model reports:
- States and Errors report
- Operating point report
- Model data report
- Error messages log report
  Errors during calculation are written to this report

A confirmation save dialog, in case of closing file when changes have been made, can be controlled using the General Options dialog conformation tab sheet.

The GSP report editor has been integrated into the model, therefore model specific. The window is dockable, so can be placed where the modeler prefers this output window.. If this window is not visible (hidden) use the Project window menu option View.



Undocked report editor window

Report toolbar

From left to right these buttons stand for:

- `New (CTRL + N)`
  Create a new file
- `Open... (CTRL + O)`
  A browse for file dialog will be opened to select a file
- `Save (CTRL + S)`
  Save the current report to file
- `Save As (ALT + S)`
  Save the current report to an alternative file
- `Print (CTRL + P)`
  Print the report

- `Cut (CTRL + T)`
  Cut text selection
- `Copy (CTRL + C)`
  Copy text selection
- `Paste (CTRL + V)`
  Paste selection on cursor position
- `Undo (CTRL + Z)`
  Undo last action

- `Find...`
  Opens a find text dialog

- `Clear report Editor`
  Clears all the text in the editor

6.6.3.1 States and Errors report

This report can be called from the `Model Data` menu from the model window. This will launch the GSP Report Editor and display a states and errors report of the current configured engine model. Note that this report requires a design calculation, in which all states and errors are determined.

6.6.3.1.1 TJET States and Errors report

The following text shows an example of the the states and errors that are reported for the TJET example model. The mathematical equation systen comprises of 4 state variables and 4 error equations.

```
GSP 11     TJET.mxl                                      12:19  July 15,
2015
STATES AND ERRORS INFORMATION
==============================================================================
===
Index indicates index in global Model State/Error arrays
Indexes "0" or "-1" indicate unused state or error

Nr. 1 Component: Inlet
------------------------------------------------------------------------------
---
Name     Index    Description
State 1      1    Exit corrected mass flow
------------------------------------------------------------------------------
---

Nr. 2 Component: Compressor
------------------------------------------------------------------------------
---
Name     Index    Description
State 1      2    rotor speed
State 2      3    map beta value
Error 1      1    mass flow - map mass flow
------------------------------------------------------------------------------
---

Nr. 5 Component: Turbine
------------------------------------------------------------------------------
---
Name     Index    Description
State 1      4    map beta value
State 2      2    rotor speed (obtained from shaft object)
Error 1      2    mass flow - map mass flow
Error 2      3    shaft power, required - delivered(delta H.W)
------------------------------------------------------------------------------
---

Nr. 6 Component: Exhaust nozzle
------------------------------------------------------------------------------
---
Name     Index    Description
Error 1      4    mass flow (in - out)
------------------------------------------------------------------------------
---
```

6.6.3.2 Model data report

An overview of the user specified model data (including component data) may be generated by selecting `Model Data|Model Data Report` in the main window. The model data report provides a way to view, export (save) or print the model data (including component data)in text format. The user specified data per component are presented in a rich text format editor which can be used to modify the layout, print or save the text in a file for subsequent processing in a modern word processor.

6.6.3.3 Operating point report

The data of the selected output parameters of the last calculated (single) operating point (either steady state or transient) can be presented in text format by selecting `Results|Operating Point Report` in the model window menu. The model data report provides a way to view, save or print the model data (including component data) grouped per component in rich text format, or output the data for subsequent processing in a word processor.
For example, to output the design operating point data to a textual report, do the following:

1. Perform a design calculation using a model with run mode 'Design'.
2. Click `Results|Operating Point Report` in the model window menu.

6.6.3.4 Error handling

GSP contains extensive error handling routines providing information in the form of error messages.

Three types of error can be identified:

- *User input errors*
  User input errors are generated usually before (or sometimes after) a simulation run when GSP detects that input is invalid or conflicting with other input or that a model configuration is invalid.
- *Simulation errors*
  During simulation, errors may occur due to conditions such as inconsistent model configurations, thermo-dynamical limitations encountered beyond the operating envelope, unrealistic transient inputs and many more. Generally, errors result from non-realistic engine model operating conditions or component parameters.
- *Results processing errors*
  These type of errors may occur:
  - at the end of a simulation calculation during post-processing of simulation result data, calculation of derived parameters, or
  - during presentation and analysis of results in graphical or tabular format for example when parameters required for graphical presentation parameters are missing.

Errors can be generated by:

- the global GSP environment and/or the gas turbine model, or
- by the code for a specific component

The error messages are divided in two categories:

- Model specific error messages
- Component specific error messages

6.6.3.4.1 Error not implemented

Help for this item is not yet implemented.
In case this error occurs frequently, please contact the GSP Development Team.

Please report the error text and GSP version number, which can be found by clicking `Help|About` in the main program window, or alternately right-clicking the file icon of the `GSP.exe` file (located in the installation directory, e.g. `..\Program Files\NLR\GSP\`), selecting `Properties` and looking in the `version` tab sheet.

6.6.3.4.2 Error status information

The right panel on the status bar gives information on errors detected by GSP during initialisation and simulation of the engine model:

- `Calc.Error`
  Indicates an error during calculations within the component models. Often these are divisions by zero or gas model iteration errors due to impossible or erroneous (component) model specifications.

- `Jac.Inv.Error`
  Indicates GSP could not determine the inverse of the linearisation Jacobian of the model during Newton-Raphson iteration towards a steady state or quasi-steady state point.

- `Iter.Error`
  Indicates GSP stopped trying to iterate towards a steady state or quasi-steady state solution due to insufficient progress. This is usually caused by excessive steps in model input. If for example the fuel flow is changed in one step from the design point value to the idle value, GSP may not be able to iterate due to the usually very non-linear gas turbine engine operating range. The solution then is to move to the idle point in a few steps or a gradual transient (or steady state series) calculation.

6.7 GSP Files

6.7.1 Project Files

.MXL
As of GSP 11 a new model format was required to store the multiple levels of model data. This format is in plain text (ASCII format), where xml is used as the storage structure. To prevent the duplication of data, child data which is equivalent to the parent node is deleted. This is the default file extension for GSP 11 project files.

.XML
Since the storage mechanism is XML, the alternative is to store projects as .XML files. Files with .XML can, when the file extension is not linked to GSP, be opened with a default xml editor (or an internet browser). On installation GSP can be linked to the XML extension as well, but remember that when using XML files for other applications, double clicking the XML file will results in opening GSP, which is not the required action. The .MXL extension is the preferred extension for GSP project files.

.MDL
This is the old GSP version 10 single model binary file format. Up to GSP 11 this was the standard model format to store (single) model data binary (note that optionally a layer from the project can be saved as .MDL). This format is a binary formatted file mainly used to save gas turbine models up to version 11 of GSP. All data set in the model components will be saved. Additional created output (e.g. simulation results) can be stored separately using text files (e.g. by saving reports), or directly to file (using Borland Database Engine to save to Paradox tables; .DB). As of version GSP 12, the old MDL file is no longer supported.

6.7.2 Map Files

.MAP
The off-design component characteristics are saved in map files. These files are in text format, and can be e.g. generated with SMOOTHC™ and SMOOTHT™ software. Off-design maps can be used for off-design behaviour of ducts, compressors, turbines heat exchangers, nozzles, inlets and combustors.

6.7.3 Output files

.FDB
Output generated by design and/or steady state (series) or transient calculations can be stored in tables. These table are stored in binary database files (.FDB).

.TXT
The states and errors report, model data report and the operating point report are output in text format and can be saved as text filed with the default TXT extension.

7 GSP models

7.1 Gas turbine models

Gas turbine engine models in GSP are represented on the model panel in the project window including a number of components models. The set of components on the model panel represents the configuration or case selected in the project tree.

Due to the object oriented structure of GSP, all models have their own environment of options, status, operating conditions, input specification data, output table and graph windows etc. That is why GSP is capable of managing different models at the same time (and copying components from one model to another). Models may also be used as component libraries.

Internally GSP keeps track of the components and their arrangement in order to set up a number of state variables. The state variables are embedded in a number of "virtual" implicit non-linear differential equations, hidden in the code of the various components (see Model theory). The non-linear equations represent the actual complex gas turbine model performance characteristics. Each model maintains its own set of equations. Different models may be open at the same time and be run intermittently, for example to compare results.

Most GSP component models use "maps" to represent multi-dimensional non-linear component characteristics. These maps are stored in text files, read into the component' s memory upon model initialization and have to be present in the specified location or embedded in the model.

7.1.1 Turboshaft models

Modeling turbo shaft engines in GSP is quite complex. The table below can be used to setup the turbo shaft model correctly.

	Operating type	GG compressor model option	GG turbine model option	FPT option checked	FPT model option	APBEq option checked
Dual shaft (GG and FPT)	Steady state load specified	free state rotor speed	free state rotor speed	☑	Speed determined by shaft (external control)	☑
	Steady state Wf specified	free state rotor speed	free state rotor speed	☑	free state rotor speed (constant)	☒
	Transient with specified load and PT governor	free state rotor speed	free state rotor speed	☑	free state rotor speed (variable)	☒
Signle shaft	Steady state load (TQ or PW) spec.	user specified rotor speed	user specified rotor speed			☒
	Steady state fuel Wf specified	user specified rotor speed	user specified rotor speed			☒
	Transient with load specified and PT governor	free state rotor speed	free state rotor speed (variable, PT governor shafts GG = PT)			☒

Nomenclature	
APBEq	Add Power Balance equation
c	compressor
exh	exhaust
FPT	Free Power Turbine
gg	gas generator
L	specified load
P or PW	power
t	turbine
TQ	specified torque
W_1	flow through the turboshaft
W_c	corrected flow through component
β	compressor/turbine map variable

7.2 Modeling theory

GSP is a component stacking modelling environment for gas turbine engines. Performance of the individual components can be simulated by component models with various levels of fidelity. As stated in the Models section, a GSP model evaluates a "virtual" set of non-linear differential equations. [1] and [2] explain more about the mathematical and numerical basics used in GSP.

Further information can be found in the GSP Technical Manual.

7.3 Component model fidelity

GSP component models may have different levens of fidelity. GSP's standard component models are all non-dimensional. Some custom component models are 1-dimensional. Non-dimensional gas turbine component models use gas properties, averaged over the flow cross areas at entry and exit stations only. For example, combustor discharge gas properties are represesented with single pressures, temperatures, flow rates, Mach numbers, densities, fuel air ratio's etc. Pattern effects are not taken into account. For general (i.e. "whole system") gas turbine performance analysis, this assumption does not produce large inaccuracies.

Note that GSP component models do not need to be limited to non-dimensional or even 1-dimensional. Due to GSP's flexibility there is principally no limitation to modelling fidelity inside a component model. In some engine specific custom component libraries, dimensional models exist, such as the 1-dimensional combustor model which calculates combustion kinetics depending on some combustor geometrical properties.

7.4 Equations

GSP calculates engine performance and gas condition changes across the components using the following equations:

- equations for conservation of mass
- equations for conservation of energy
- the perfect gas state equation
- the isentropic flow equation
- equations for conservation of momentum of gas flow
- the equation for rotor inertia effects
- equations for heat flux between the gas path, material and ambient environment.

From these equations, a set of non-linear differential equations (NDEs) is arranged and solved by the GSP solver. Since gas turbine off-design models are particularly non-linear, customary solvers often fail to converge. Therefore GSP has it's own Newton-Raphson based solver optimized for gas turbine models. The model operating point is defined by a number of states (or 'free states') collected in a state vector. The number of NDEs equals the number of states and the deviation from a valid solution is represented by the error vector which holds the error values. The GSP solver iterates towards the solution where all errors are zero (within the user specified tolerance).

7.5 States and errors

For solving the GSP non-linear differential equations (NDEs) GSP defines the operating point by states (or 'free states') in a state vector. Using the appropriate aero-thermodynamics equations, maps and other relations, all engine parameters can be directly derived from the states. As such, the states represent the unknowns in the NDE set to solve for. The NDEs are depending on the state vector and each NDEs has a error variable representing the deviation from a valid solution. The GSP solver iterates towards the solution where all errors (i.e. the error vector) are zero (within the user specified tolerance). Note that the NDEs can not be considered equations that can be represented by a series of mathematical expressions (functions of state variables) but rather represents the outcome of several algorithms using thermodynamics, table/map look-ups, internal iterations etc. This is what makes the non-linearity so significant.
For a simple turbojet model (tjet.mdl) for example, there are 4 states and 4 errors. For more complex models such as turbofan models with several schedulers, the number of states and equations may easily rise up to 20 or more.

Although most states and errors are set up automatically by GSP, the user can have control over states and errors using component model options. The current maximum amount of model state variables is set to 50. Please contact NLR when your model exceeds this maximum to discuss options.

7.5.1 Controlling states and errors

Although most states and errors are set up automatically by GSP, the user can have control over several states and errors using component model options or special control or scheduling components. For example, by adding a Rotor speed controller to a model, the user extends the set of equations with equation requiring a rotor speed to be equal to a user specified value. The user then also has to add a state variable to make the number of states and errors equal again. Usually with a Rotor speed controller, the fuel flow input (or other Power control input) is turned into a state.

If the `Show advanced model equation controls options in components` Advanced Project option is active, the rules for creating error equations and state variables can be controlled as well as state variable numbers assigned to deactivate particular error equations.

7.5.1.1 Always create error

If the `Show advanced model equation controls options in components` Advanced Project option is active, the `Always create error` checkbox is shown on components that are able to add an error equation to the model such as several of the Power control and Scheduling components. With this option active, an error is always added to the model to ensure the same equation set is used in a case regardless of the component being active or not (for numerical stability reasons or in cases where the model is externally controlled using the GSP API). If the error equation is added but the component is not active (unchecked `Active` check box on general tab sheet), a dummy equation is used instead to maintain numerical consistency using a state with a number specified in the `Corresponding state nr. for deactivation` field. Usually the corresponding state is a control input with the the `Always create state` option set.

7.5.1.2 Corresponding state nr, for deactivation

If the `Show advanced model equation controls options in components` Advanced Project option is active, the `Corresponding state nr. for deactivation` checkbox is shown on components that are able to add an error equation to the model such as several of the Power control and Scheduling components. When a number larger than 0 is entered, the corresponding state number is used for the Dummy equation when the Always create error option is checked.

7.5.1.3 Always create state

If the `Show advanced model equation controls options in components` Advanced Project option is active, the `Always create state` checkbox is shown on Control components that can be configured with the input as as state variable to match an extra error equation added by other control or scheduling components. With this option active, a state is always added to the model to ensure the same equation set is used in a case regardless of the component being active or not (for numerical stability reasons or in cases where the model is externally controlled using the GSP API). This setting usually needs to be accompanied by a `Always create error` option set in a component adding an error equation.

7.5.1.4 Dummy Equations

A GSP model equation may be replaced by a dummy equation if the actual equation is deactivated (e.g. by unchecking the Active field in the General tab sheet of a controller component). The dummy equation must depend on at least one state variable in order to prevent a singular Jacobian matrix that cannot be inverted. This means the dummy equation needs a corresponding state (that has to be dummy as well) and not controlled anymore by the iteration solving the set of equations. The corresponding state often is user specified. As such this state variable is not connected to the internal model parameter (such as a rotor speed or something else) but simply used to form a dummy error equation of the form Error = State-1. This results in the particular dummy state to become equal to 1 and the error equal to 0 upon convergence without interfering with the engine model equations.

7.6 Properties

GSP model properties are model parameters that can be controlled from outside during a simulation for example by Scheduling components. Properties represent internal component parameters and adaptation of properties requires understanding of the GSP internals and is therefore advised for advanced users only.
Note that changing properties during simulations may have unexpected results. There may be cases where the property change may interfere with iterations and cause failure to converge. In other case the change may not have any effect because GSP somewhere resets its value depending on other parameters.

7.7 Heat transfer

In GSP there are several methods by which heat transfer among the various components and the surrounding environment can be modeled:

- A steady-state heat flux can be specified in a Duct component. A negative value will stand for a heat flux out of the system, whereas a positive value represents a heat input. For this component a heat flow is defined specifically and cannot be related to the system in the cycle calculations.

- A steady-state or dynamic heat flux using the Heat Sink component. The Heat Sink component has the ability to simulate heat transfer via conduction, convection and radiation among components and with the ambient environment.
- A Heat soakage heat flux during a transient, heating up or cooling down the gas path surrounding walls until thermal stability has been obtained (average wall and gas temperatures equal). Heat soakage has an effect on performance, for example slowing down an acceleration by absorbing part of the combustor released heat for heating up the combustor and/or turbine walls.

Compressor and turbine performance is significantly affected by heat transfer causing the process to become non-adiabatic. In GSP this effect is modeled by dividing the total heat flux (i.e. the sum of heat sink and heat soakage heat fluxes) in two heat fluxes, one happening before (start heat flux) and one after (end heat flux) the compression or expansion process. The compression or expansion process itself is happening 'in between' the points of the start and end heat fluxes and is calculated as adiabatic. The start heat flux increases the compression or expansion start temperature thereby affecting compression (increase if heat is added) or expansion (decreased in case of heat loss) work. The en heat flux does not affect the work but merely changes the exit enthalpy (and thus temperature).
The ratio of the start heat flux to the total can be user specified in the compressor and turbine Design tab sheets.

7.8 Component off-design performance

Off-design component performance is determined by component characteristics which often are maps (e.g. compressor or turbine maps) or equations (such as simple pressure loss equations). Off-design performance is defined *relative* to the design point. This means off-design performance is scaled proportionally with the design point data. This mechanism is best demonstrated using map scaling.

7.9 Dynamic effects

For transient calculations, dynamic terms have to be included in the differential equations to simulate gas dynamic effects. In GSP the following dynamic effects are taken into account:

- Rotor inertia effcts (on accelerations and decelerations)
- Heat soakage effects (the component material is a "heat sink" during gas temperature variations)
- Volume dynamics effects (compressible gas mass storage capacity in components affect transient performance).

7.9.1 Rotor inertia effects

The spool inertial moment specifies the total moment of inertia of all elements connected to a shaft including turbines, spools and compressors, and can be specified in the `General` tab sheet of the turbine component.

7.9.2 Heat soakage effects

Gas path components for gases with potentially high temperatures allow specification of heat transfer data to allow simulation of heat flow to or from the component material during transients. During an gas turbine engine acceleration for example, the turbine material is heated up to the new steady state temperature level and since this heat is taken from the gas, transient performance may be affected significantly. The reverse case happens during

deceleration when the material heats up the relatively cool gas. Heat soakage heat transfer coefficient is a function of and gas path conditions and user specified data, to be specified in the 'Heat soak' tab sheets of gas path components data entry windows. Note that also the Heat sink will cause a heat soakage effect (on top of the steady-state heat transfer) if the heat sink mass is specified >0.
The material temperature dynamic response is a 1st order response following the equation:

$$Q = U_{ht} \cdot A_{ht} \cdot (T_g - T_m)$$

with heat transfer coefficient U_{ht} calculated as follows:

$$U_{ht} = \frac{1}{\dfrac{1}{FC} + \dfrac{l_{eff}}{k_m}}$$

FC is the film coefficient, calculated from

$$\frac{FC}{FC_{des}} = \frac{c_p}{c_{p,des}} \left(\frac{W}{W_{des}} \right)^{0.8}$$

with FC_{des} as a user specified film coefficient in the design point. $c_{p,des}$ and W_{des} are the gas specific heat and mass flow calculated for DP and c_p and W the OD calculated values.

The average material temperature T_m is calculated by integrating:

$$\frac{dT_m}{dt} = \frac{Q}{c_m \cdot M_m}$$

Initial value for T_m is the steady state gas temperature. Since heat flow is assumed to be proportional to the difference between gas and material temperature the material temperature T_m follows the average gas temperature T_g with a first order lag. The time constant is determined by

$$\tau = \frac{c_m \cdot M_m}{U_{ht} \cdot A_{ht}}$$

Symbols	Description	Parameter name
T_m	average material temperature	
T_g	average gas temperature	
A_{ht}	effective contact surface	Heat soak A
C_m	specific heat component material	Heat soak Cp
M_m	Effective mass	Heat soak mass

U_{ht}	Heat transfer coefficient	Uhsoak
L_{eff}	effective length heat flow transport	Heat soak Leff
k_m	material thermal conductivity	Heat soak k
FC	Film coefficient	Heat soak film coefficient
W	Gas mass flow	
C_p	Gas specific heat	

For more information, see the GSP Technical Manual.

7.9.3 Volume effects

Most gas path components allow specification of volumes to allow calculation of volume effects during transients. Volumes in the gas path act as buffers capable of temporarily storing extra gas due to relatively large pressures during an acceleration for example. Components with relatively large volume such as recuperators can have a significant effect on transient performance.
Note that large volumes may easily lead to unstable system operation and therefore also to numerical convergence problems in GSP. Therefore, volumes often need to be specified *between* components and in GSP this is done by adding a duct component with a volume.
See the Technical manual for information on the theory and equations of volume effects.

7.10 Atmospheric models

In the flight conditions window the gas turbine ambient operating conditions, which affect model performance through components such as the Inlet and the Exhaust components can be specified.

Three model types are available to determine the relation between altitude, air speed and air temperature and pressure:

- `ISA`
  The International Standard Atmosphere with a fixed altitude - pressure - temperature relation. Pressure altitude parameter name in GSP is 'Zp'.
  The implemented ISA from ICAO supports altitude - pressure - temperature ranging from -5 km to 80 km (see ref. [8]).
- `ISA+`
  As ISA, but with a user specified deviation `dTs` from the standard temperature
- `Custom`
  User specified pressure and temperature independent of altitude

Air speed can be specified in:
- `Mach`
  Flight Mach number
- `Vt`
  True air speed in m/s
- `Vc`
  Calibrated air speed in m/s

The resulting total pressure `Pt` and temperature `Tt` are automatically calculated.
Density is not used in GSP calculations.

7.10.1 Ambient/Flight Conditions window

The ambient/flight conditions window provides control over the gas turbines ambient operating conditions, which affect model performance through components such as the Inlet and Exhaust components.

Flight or ambient conditions can be specified according to three different atmospheric models.

The ambient/flight conditions window contains the following tabs:

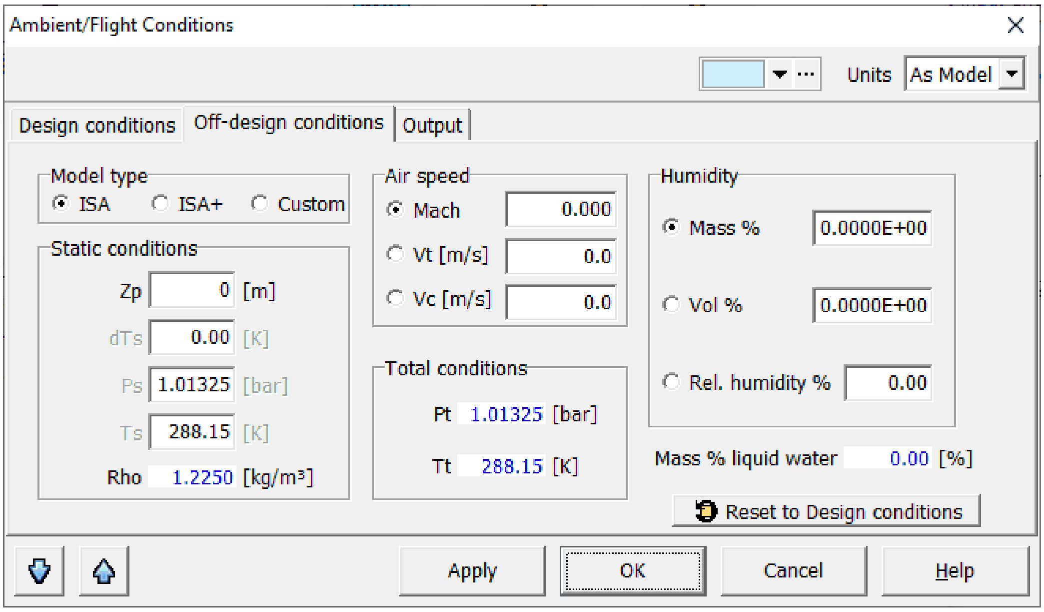
- `Design conditions`
  Reference design flight or ambient conditions are specified here for which the engine is designed.

- `Off-design flight/ambient conditions`
  Steady state off-design flight or ambient conditions are specified here.

- `Transient or St.St. OD Series`
  Off-design steady state series or transient flight or ambient conditions as a function of time are specified here in a transient input grid. To input or change variable data at a time step consistent with the selected atmospheric model, editing is not performed in the grid but in a seperate specification window, which pops up after double-clicking one of the variables or by clicking the `Edit` button. After having edited variable data, click the `Apply` button to recalculate all variable data for the time step.
  Note: GSP can not interpolate values between different atmospheric types; therefore, be sure to use the same atmospheric type (i.e. ISA, ISA+ or Custom) for all transient input points or your transient simulation will not work (values will remain constant).

- `Output`
  Use this tab to select the output parameters to populate the output tables. Note that there are options that output the static atmosphere data (see the `Static conditions data` options group), other groups of options output the flight conditions or total parameters (`Total conditions data` and the `Air speed data`). The last group of options (`Gas/Air composition`) output composition property data parameters.

- `Humidity`
  Design and off-design ambient humidity can be specified as a mass percentage of the total flow (`Mass %`), as a volumetric percentage of the total flow valid for the vapour only (`Vol %`), or as relative humidity (`Relative humidity %`). The liquid water mass percentage (`Mass % liquid water`) is determined in the calculation.



7.10.2 Pressure Altitude

It is common in dealing with airborne research data to encounter many different altitude terms. These include geometric altitude, GPS altitude, INS altitude, pressure altitude, geopotential height, and so on. Despite the nomenclature, there are only two altitude scales involved: *geometric altitude* and *geopotential altitude* or height.

Geometric altitude
Geometric altitude is the scale we are most familiar with; it is what we would measure with a tape measure.

Geopotential altitude
Geopotential altitude is based on a scale that relates altitude to gravitational equipotentials, or surfaces of constant gravitational potential energy per unit mass. Although geopotential altitude approximates geometric height, they are not equal. An important type of geopotential height is *pressure altitude*, which is based on a standard atmospheric model for temperature as a function of pressure. One particular model, the International Standard Atmosphere (ISA), is what all aircraft altimeters use to relate static pressure measurements on an aircraft to a corresponding pressure altitude scale. There are also a number of additional altitude terms related to flying airplanes such as true altitude, indicated altitude, absolute altitude and density altitude.

Pressure Altitude
Although pressure altitude is a type of geopotential height, it is treated separately because of it's importance in atmospheric research. The pressure altitude scale is based on the International Civil Aviation Organization's (ICAO) International Standard Atmosphere (ISA) (see ref. [8]). It can be used to establish, using the hydrostatic equation and the ideal gas law, a relationship between pressure and pressure altitude, using geopotential height. It differs from "normal" geopotential height in that it is based on a model and it assumes that the humidity is zero. The model seldom looks like the actual atmosphere a plane is flying in, and real atmospheres never have zero humidity.

Pressure altitude is used so that aircraft, which use static pressure to determine altitude, can agree upon what "altitude" they are flying at without having to continually update their altimeters with local pressure corrections. Technically, this is only true above 18,000 feet (FL180). Below this altitude in North America aircraft make local altimeter corrections to ensure that they are flying at the correct altitude. Using this definition for pressure altitude, a pilot can say "I'm at Flight Level 330." (that is 33,000 feet), instead of "I'm at 262 hPa." Pressure just isn't very intuitive since it's logarithmic with altitude, and it also decreases with altitude. In addition to pressure altitude, there are five additional altitude scales relevant to aviation.

Relation between geometric altitude and geopotential altitude:

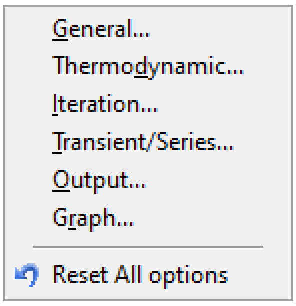
$$h = (Re * H) / (Re - H)$$

where Re = 6 356 766 m, the nominal radius of the earth
H = geopotential altitude, m
h = geometric altitude, m

The parameter name for geopotential altitude (H) or pressure altitude (since we use the ICAO ISA) in GSP is 'Zp'. The SAE ARP 755 standard parameter name is 'alt'.

7.11 Model options

In the `Options` menu all options which apply to the model in the selected configuration or case can be specified:



- `General`
  Specify general options like enabling confirmation dialogs.

- `Thermodyn. model`
  Specify thermodynamic gas model options like real gas corrections.

- `Iteration`
  Specify iteration specific options like accuracy and numerical settings.

- `Transient/Series`
  Specify transient or steady state control parameters like start time and timestep.

- `Output`
  Specify output options like automatic table pop-up and gas composition output.

- `St.St.Graph`
  Specify graph options for the Steady State results graph window like scaling and titles.

- `Trans.Graph`
  Specify graph options for the Transient results graph window like scaling and titles.

7.11.1 General options

The `General` options menu contains the following:

Model

- `Show alternative station string in model window`
  The station designation standard has converged to a minimum of a 3-digit station designation, the use of floating point (point/decimal seperated) nimbers is deprecated. Enabling this option allows to modeler to show the alternative station designation from the output options.

Input

- `Transient Input Data`
  Press the buttons to (de-)activate all transient input

- `Deterioration`
  Press the buttons to (de-)activate all deterioration input.
  Press the "`Clear value in all components`" button to clear all deterioration input data (resets deterioration value of modifiers to zero)

Outnput

- `Confirm Grid Row Delete`
  If this check box is selected, the program will ask for confirmation before a row is deleted in a data specification grid (e.g. a transient input grid).

- `Confirm Table Row Delete`
  If this check box is selected, the program will ask for confirmation before a row is deleted in an output table.

- If changed, show save dialog for
  - `Steady State Output Table`
  - `Transient Output Table`
  - `Report editor`

Units

Global model unit system selection for the selected case or configuration in the project.
- `SI`
- `Imperial`

Note that the unit system settings in terms of formats, scale factors and unit strings can be user adapted for the predefined parameter types in the current case or configuration Output options.
The Unit system can also be set per component using the Component unit system setting; see the Component unit system for more information on the unit system.

Initialization

- `Design Point Calculations`
  - `After Design Point iteration...`
    - `Reset (change) DP input to DP iteration end result`
      This option will copy iteration results (after a design calculation) to the DP input fields. Reset (change) Design Point input to Design point iteration result after the DP iteration has been completed. This only applies if DP calculation involves the solving of an equation such as when using the DP equation component or selecting SOT input in the combustor or Design Effectiveness input in the recuperator component.
      - `Confirm DP input reset`
        Optional confirmation for enabled `Reset (change) DP input to DP iteration end result.` Confirm reset of the Design point (DP) input to DP iteration Reset DP input to DP results with an Ok/Cancel dialog box.
  - `Automatically initialize for OD calculation`
    Finish Design point (DP) iteration with initialization of OD equations (set OD states, errors and map scaling).

- `Calculate map scaling factors in Design Point cases`
  For Design (DP) cases, OD map scaling factors do not need to be calculated, only in OD cases during the pre-OD initialization DP calculation.
  However, you can check this option if you want to enforce the calculation of the scaling factors for example to assess the map scaling factors in
  a specific design point.

- `Off-design calculations`
  - `Confirm DP calculation before OD calculation`
    Confirm whether to do a design point (DP) calculation prior to an off-design calculation if no DP has been calculated yet.
- `Auto reset OD input to DP after completion of DP case calculation`
  All off-design input will be reset to design point input at design point calculations run from Design or Design Series case types. This option is beneficial if a large number of transient calculations or steady state parameter sweeps have to be performed with an initial operating point at or near the design point either in the same case changing case type, or in a child case of the DP parent.

- `Miscellaneous`
- `Reinitialize model on case type change`
  Reinitialize model when changing the case type in the project tree. Disabling this options allows simulations starting at model states left by other simulation modes such as a transient following a steady-state calculation for example.
  - `Warn for unlinked components`
    Warns for unlinked components after loading case or config model when checked.
  - `Warn for deprecated components`
    Warns for deprecated components after loading component input data when checked.

Case Management
- `Set controlled parameters back to original after simulation run`
  Maintain all original model input values after simulations. Without this option, the model input values stay as last set by case control component.
- `Warn on loosing embedded map`
  On loosing an embedded map caused by user interaction a warning will be displayed, cancel to quit loosing the map.

7.11.2 Thermodynamic options

The `Thermodynamic model` options window provides control over the following options:

`Gas model`
- `Real gas effects map correction`
  When checked, the component map parameters are corrected for real gas effects (including variations of the gas constant R and gamma), when unchecked, correction is performed with temperature and pressure only.

Corrected or reduced parameters are often used for representation of component characteristics and overall engine performance. The off-design component maps often use corrected parameter groups to represent corrected mass flow and corrected or referred speed. Full correction (dimensionless) includes the diameter of the gas turbine and the gas properties. When a single gas turbine is considered, the diameter is not important. The gas properties are important when the real gas effects are considered since gamma is dependent on the temperature and R is dependent on the gas composition.

A derivation for the corrected spool speed and the corrected mass flow shows the difference between real gas corrections and ideal gas corrections:
- Corrected speed
- Corrected flow

7.11.3 Iteration options

The `Iteration` options menu provides control over the iteration processes towards a solution for a steady-sate or quasi-steady state (transient) operating point:

- `Accuracy`
  This is the accuracy required for the solution of the set of virtual non-linear equations to be solved in GSP (it is the maximum tolerated value for each error variable). This accuracy can be considered to be roughly equal to (or of same order) the other calculated performance parameters (default 0.001).

- `Relative state perturbations for jacobian calculations`
  The relative perturbation of the state variables used to determine partial derivatives for the Jacobian matrix (default 0.001). It is a means to control the relative range across which the model is locally linearized. For stable iteration calculations the absolute value of this variable must be equal or smaller than the accuracy.

- `State correction step factor`
  This is a factor to increase or decrease the stepsize GSP uses to iterate towards the solution using the inverse Jacobian of the Newton-Raphson method (default 1).
  It is a means to avoid too large steps which cause the solution to become unstable due to non-linearities.

- `Relative state variable correction limit`
  This parameter limits the maximum relative correction of a state variable during iteration (default 0.05). It is a means to avoid too large steps which cause the solution to become unstable due to non-linearities.

- `Convergence test vector`
  This factor represents the factor GSP uses to determine whether there is conversion progress (default 0.9). The default is 0.9, meaning that GSP requires at least 10% less of the total square of the error variables per iteration step before the inverse Jacobian is updated.

- `Max. nr. of Jacobian calculations`
  This is the maximum number of times the Jacobian is recalculated per operating point calculation, i.e. per time step for transients (default 50). It is a means to stop the iteration process if there is little chance of finding a solution. When the default of 50 is encountered it is often useless to extend te limit. Instead, improving the model or control system consistensy, trying to take smaller steps towards/between new succeeding operating points or adaption of the other iteration parameters is often more successful.

- `Max. nr. of Boyden updates`
  This is the maximum number of times the Jacobian is updated (using the Broyden method) per operating point calculation, i.e. per time step for transients (default 0). It is a means to fine-tune optimum iteration speed by taking the best trade-off between recalculating the Jacobian completely which consumes much computation time and accepting an approximate Jacobian using Broyden with little computation time.

- `Iteration step limit factor`
  This factor limits the total number of iteration steps by the number of model state variables and this factor (default 50). It is a means to stop the iteration process if there is little chance of finding a solution (see Max. nr. of Jacobian calculations).

7.11.4 Transient/Series options

The `Transient/Series` options menu controls the defaults for the time parameters values and frequency of transient or steady state series calculations:

`Series control`

- `Default start time`
  The default transient start time (default 0.000).
- `Simulation step`
  The calculation (time) step or interval of calculated quasi-steady state or steady state series operating points in seconds (default 0.050).
- `Output interval`
  The time interval in seconds at which the parameter output is send to the output table (default 0.200).
- `Output at adapted points/time steps`
  When the solver changed the intervals during calculation (for numerical stability) output data at adapted points/time steps will be displayed.
- `Maximum point/time`
  The time in seconds at which the transient or steady state series must stop (default 1E20, to make the calculation proceed until a transient input time series ends).
- `Always start at default start point/time`
  Always reset start time/point to default start time value prior to series/transient simulation run.
- `Show start point/time dialog`
  Show start point/time dialog prior to series/transient simulation run.

`Reset State on each step`

- `Design point series`
  On design point series runs, reset model state variables back to 1 before each step. In some cases with DP equation controllers, this may improve convergence speed.

`Error handling`

- `Batch mode; continue without user response (report error in output table)`
  On error continue without user interaction.
- `Reset state to Design after error`
  After error reset the state to design.

7.11.5 Output options

The `Output` options menu item provides control over what output GSP generates and whether confirmation is asked for different types of simulation:

`Output Parameters`

Central place to set (most of) the output option simultaneously or per component

- `Clear all output parameters for`
  Clear the options for either `All Model Components`, `Flight/Ambient Conditions`, or `Global Outp`. Note that this cannot be undone!
- `Set output parameters for all modeling components (if option exists)`
  Three tab sheets are available to centrally set the output parameters for `Components`, `Flight/Ambient Conditions` or `System Performance`.
  - `Components`
    Depending on the selected component, all gas path components or all turbo components (specify in the combo box in the lower left corner) the output will be changed accordingly to the selected options when the apply button (in the lower right corner) has been pressed. Note that changes cannot be undone! Note that the Map `oper.crv.par.s` option will not grey out the corresponding map options as in the regular

component output data input windows due to class/object and inheritance issues. Functionally this option actually sets the corresponding output options, but does not grey it out in the central component output window!

- Flight/Ambient Conditions
- System Performance

Specify overall system performance output parameters:

- Total; *total/overall engine parameters*
- total fuel flow Wf (for example the total of primary and afterburner fuel flow)
- total intake airflow W (sum of all intake airflow values)
- total exit area A (sum of all exhaust exit cross sections)
- OPR, Overall Pressure ratio defined as Pt3/Pt2 (Combustor total entry pressure/First compressor total entry pressure)

Thrust; *typical jet engine parameters:*

- total Ram Drag RD, total ram drag calculated from all inlets, usually used to calculate total net thrust
- total gross thrust FG, the total of all exhaust nozzle's gross thrust values
- total net thrust FN = total FG - total RD
- thrust specific fuel consumption TSFC = Wf / FN (SI: [kg/(N h)]; Imperial: [lb/(lbf h)])
- installed performance parameters

Shaft power; *typical turboshaft engine parameters:*

- total shaft output power (output parameter name is PWshaft), PWshaft is the total of ALL engine/model system shaft power outputs
- (shaft power) specific fuel consumption SFC = Wf / PWshaft (SI: [kg/(kW h)]; Imperial: [lb/(hp h)])

Output Tables

- Output to case tables active

  Enable/disable output to table

General

- Add case node number/name to comment

  Adds the node number, or the name of the simulation case for quick reference in front of the comment
- Add user specified text to comment

  Adds a user specified comment for quick reference behind the comment text, select a custom string or a timestamp.
- Curve numbering
  - Treat <case-break> as <group> in renumbering

    Use the <case-break> to count as a <group> break when renumbering multiple simulation case data tables
  - Restart curve number counting on <group> break

    Restart the curve numbering when a <group> break is found in the data table

Design Point

- Output to table

  Check to output the DP calculation results to the output table
- Confirm single point DP table output

  Confirm table output at single Design point calculations
- Confirm initialization DP table output

  Confirm table output at Design point calculations that have been completed to initialize off-design calculations.
- Automatic table pop-up

  Table window receives focus when the calculation finishes so that the window become visible.
- Add break between Init. Design and St.St.

  Automatically add break after the initial Design point calculation table output preceding the Steady-State output.

`Steady State`

- `Output to table`
  Check to output the OD St. St. calculation results to the output table
- `Confirm single point table output`
  Confirm table output at single OD / Steady-state point calculations
- `Automatic table pop-up`
  Table window receives focus when the calculation finishes so that the window become visible.

`Transient/Series`

- `Automatic table pop-up`
  Table window receives focus when the calculation finishes so that the window become visible.
- `Add break before new series`
  Automatically add break before output of a new St.St. Series or Transient starts

`Default Comments in tables`

Select to insert a default comment per output data row for Design, Steady State, St St. Series, and/or Transient.

`Output Parameters`

- `AS755 Station numbering`
  This contains the interface to a lookup table to map the older standard floating point station numbers to the current standard 3-digit station numbers.
  The `Alternative station number` checkbox controls whether the alternative station is used in output parameters or not. For display of the 3-digit station number, General options
-

Gas composition

Specifies output for each component in the gas composition. Output of the gas composition can be in mass percentage (`Mass%`) or volumetric percentage (`Vol%`). Check specie to add this specie to the output table when the output parameter to output gas composition is checked on the `Output` tab sheet.
Check `add FAR (Fuel Air Ratio)` to include the FAR output parameter. Note that for showing the FAR parameter the `Gas comp. in` or `out` option of the component property window has to be selected.

Oper.Point report

`General`

These options control whether and how output is sent to the Text output report window:

- `Automatic Report pop-up`
  Pop-up when finished writing to the report editor
- `Omit no-output components`
  If checked component output is ignored when no output parameters are selected for that component in order to avoid component headers without output data.
- `Auto-clear existing text report`
  If selected, the report text is cleared before every new calculation.
- `Append to existing text`
  If selected, the report text is appended to the existing report text for every new calculation.
- `Confirm saving report on exit`
  If selected, a confirmation dialog is shown to save the report when closing a project.

`Output data groups`

Options to include/exclude data in operating report, or how the data is formatted

- `Ambient/Flight conditions`
  Include the Ambient/Filght condition parameters
- `Station oriented output table (W, Tt, Pt, Wc per engine station)`
  Output data formatted per station (this is the default setting)
- `Component oriented output table (all data per component)`
  Output data formatted per model component

- Global system performance
  Include system performance parameter data
- Calculated Expressions
  Include the calculated expressions results

Calculated expressions

This list sums the definitions for added calculated columns. The expressions are saved with the model and are automatically recalculated after re-opening a model and running simulations. If the expression cannot be calculated (for example because output fields have been deleted or definitions/component nrs. changed), the status field indicates it as invalid and the expression requires adaptation. Otherwise OK indicates a successfully evaluated expression. Use this interface to edit/create/delete calculated expressions.

Debug

- Debug output options
  Advanced feature to display information on the iteration process, useful for models having difficulty in reaching convergence.
  - Debug output after error
    Do enable debug output info to the Report editor
  - Confirm Debug output
    User confirmation to output the debug information
  - States, Errors during iteration
    Do output the state/error values during iteration
  - Jacobian's during iteration
    Do output Jacobian matrix during iteration
- Error log options
  - Log errors in model text report
    Do log all errors in Report editor
  - Automatic error log pop-up
    Do pop-up the Report editor
- Stop on and report....
  - Chem. Equilibrium calc. errors
    Terminate calculation and report error

Output Identification

- Derive parameter ID from
  Component based output parameter names include the unique name of the component or the component ID (number).
- Warn on duplicate output parameter ID's
  Warn if duplicate parameter ID/name found, such as TT2 coming from both and Inlet (exit station 2) and Compressor (entry station 2).Normally the parameter values are identical so this option can safely be unchecked in most cases.

Formats

GSP has specific formatting, scaling rules and unit strings for a list or predefined parameter types and both SI and imperial units (set by the Units settings in the General options or per component in the Component unit system), covering most in- and output parameters listed in the Output options Formats tab sheet (click Reset to standard if the table is empty). These predefined settings can be user adapted. E.g. when modeling small gas turbines, or small bleed flows more digits than the default may be required. For micro turbines one would like to use g/s (grams/second) instead of kg/s for mass flows and change the scale factor to 0.001. See format description for an explanation on parameter formatting strings.

- Use custom formats
  Activate this option to activate the adapted formatting, scaling and unit strings.
- Reset to standard
  Press button to reset the table to default values, or in case of an empty table populate the table with the default settings.

7.11.6 Graph options

The `Graph` options dialog window offers control over the series output graphs. The Graph options window can always be opened from the Graph window. From the Options menu in the Model window the graph options for steady state and transient output graphs can also be specified.

General

Titles

- `Title 1 / 2`
  Displays graph title at top-center above graph.
- `Number curves`
  If checked, markers are displayed along the curve at each datapoint.
- `Auto-Refresh`
  If checked, the graph is automatically redrawn at each variable or option change. A manual refresh can be triggered by clikcing the `Refresh` button left-below the graph.

Gridlines

For the X-axis grid and Y-axis grid the following options are available:

- `No grid`
  Omitts all grid lines.
- `On Maj. Tics only`
  Draws grid lines for major ticks only.
- `On Maj. + Min. Tics`
  Draws grid lines for all ticks.

- `Font`
  Brings up standard font change dialog in which font properties can be set.

Axes

For the x-axis and all 4 y-axes the following options are available:

- `Autoscaling`
  If this check box is selected, all axis will be scaled automatically by GSP according to the minimum and maximum values in the dataset. This may cause longer time before displaying with very large datasets.
- `Label`
  An optional text located along the appropriate axis, replacing the standard GSP denominations.
- `Min`
  The minimal axis value.
- `Max`
  The maximal axis value.
- `Maj Tic`
  The number of tick marks inbetween the minimal and maximal value
- `Min per Maj`
  The number of tick marks inbetween two major tick marks
- `Exponent`
  Additonal exponent variable along axes for scientific notation
- `Decimals`
  The number of decimals for values along axes.

`Line Styles`

Set for a maximum op 8 different lines through selecting the appropriate line:

- `style`
- `width`
- `color`
- `marker symbol`

A symbol set at each calculated point.

`Legend`

- `Legend parameter ("Z") to identify multiple curves (select None to deactivate legend)`
  Select the Z parameter

Legend position and off-set can be customized for values and legend.

`Carpet`

- `Draw carpet lines`
  Enable drawing of carpet lines
- `Carpet lines in next color`
  Use different colors (specified in line-styles tab)
- `Carpet parameter ("Z2") (select None to deactivate legend)`
  Select the Z parameter

`Groups`

- `Repeat legend values for subsequent groups`
  For subsequent groups of series of rows allow legend value drawing
- `Repeat legend labels for subsequent groups`
  For subsequent groups of series of rows allow legend label drawing

`Security`

Select the security markings for the graph using the drop-down boxes.

8 GSP components

8.1 Components general

Components are the main building blocks for building GSP models. In the GSP interface, a component is represented by an icon on the model panel, symbolizing the specific gas turbine component.

Three main categories of components can be identified:

- Gas Path Components,
- Control Components,
- Custom Components.

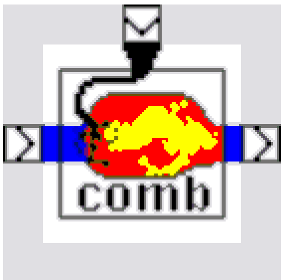
Components are drag-and-dropped from library windows onto the model windows and arranged to form the desired gas turbine configuration. Once a component is on a model panel, its properties and characteristics can be specified using the data specification window (accessed by double-clicking the icon).
Instead of representing a gas turbine model, model windows can also be used to hold a number of (unlinked) components and serve as a repository of pre-configured component models to be saved and copied to other models.

Refer to the Technical Manual for a comprehensive description of the component models.

8.1.1 Component icon

A component icon in the model window represents a gas turbine component integrated in an engine model. Double-clicking the icon brings up a component editing form. The right mouse-button brings up a pop-up menu for the component editing form an an option to rotate the icon 90 degrees.
Most components are linked to others using the small link icons representing either a control or a gas flow link. Some components are unlinked and usually represent subsystems defining some global model variable such as compressor bleed flow or turbine external load.

Click on the items in the example icon (a combustor) below to learn more about component icons.



See model window to see an example of icons linked together forming a gas turbine model.

A descriptive fly-over hint for model components, when placed on the model window , now includes the user defined Name and ID string; can be extended with anything in the future.

Double-clicking will open the Component data window, right-clicking will open the component pop-up menu.

8.1.1.1 Component number

The component shown on the upper right of a component icon number identifies the component model in the model window. After model initialization, these component numbers indicate the order of calculation during the iterations towards valid DP and OP operating points. The component numbering is automatic and 'dynamic' in a sense that they change after adding or deleting components in the model panel. Unlinked (non gas path, non control system etc.) components can be manually moved upwards or downwards in the sequence to change the calculation order. This may be necessary in some cases, for example:

- when determinate relations in schedulers are required to be evaluated before others
- additional output parameter component expressions use the result of other additional output parameters; then the latter should be moved before the former.

The component numbers are also shown in the (read-only) top right Calculation order field in the component data windows.

Because GSP internally uses the component numbers for ordering components in the iterations, the user cannot modify component numbers.
Note that if the component D string is empty, the component number is used instead to identify component parameters.

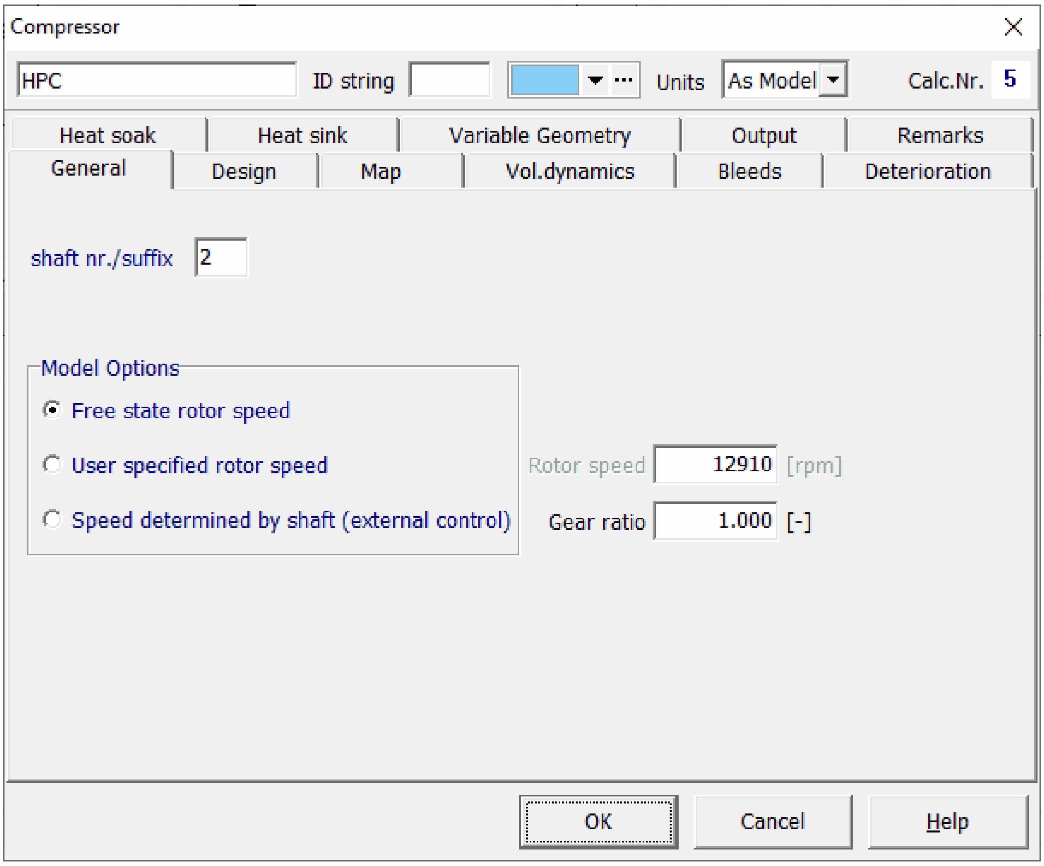
8.1.1.2 Link icon

⊡ (this is an example, rotated images can be found on certain components)

GSP uses *Link objects* represented by *link icons* to facilitate interaction among linkable components. The link elements are visualized by small rectangles on the component icons. The little chevrons inside indicate the direction of the gas flow or control information. Unlinked link icons are white. The color blue is used to indicate established links between components, representing the gas path between the components (also the little chevrons [>>] turn yellow). Black is used to indicate the link is between a component and an associated control system component, such as a fuel control component for the combustor.

8.1.2 Component data window

Double-clicking a model component (or icon) from the model window will open the data entry or input window of the selected component. Alternatively right-click the model component to display the component pop-up menu, and select `Edit` (alternatively press Alt and E key together for a selected or selection of component(s)). A window similar to the picture below will be shown. The window displays various tab sheets where the modeler can setup the specific data for the selected component. The figure below shows an example of the compressor data input window.



The component data window has a layout consistent for all components. At the top the general component configuration and identification data are displayed:

- `Component name`
  The user-changeable name used in the textual output to identify the component
- `ID string`
  If user-specified, this string will be added to tabular component output parameter names, replacing the component number
- A color selection interface to determine the background color in the output data table
- The `Units` field to select the Component unit system
- `Calc Nr.`
  The component number as determined by GSP during model initialization.

Depending on the component, several tab sheets can be selected of which the most common are:

- `General`
  The `General` tab sheet is used to set general component properties, off-design input parameters, component specific data and component options. After a design point has been calculated, the `General` tab sheet will usually be used to change component properties for analysis of off-design behaviour.

- `Design`
  Design values, always colored navy blue when using standard Windows colours, are used for specification of the component's design point performance parameters. Use this tab sheet only prior to a new design point calculation. Design values are not used during off-design simulation if corresponding variables are specified in other tabsheets. It should be

remarked that running a design calculation resets off-design variables to their design variables, if `Auto Reset Input to Design` is checked in `Options|General`.

- `Map`
  The `Map` tab sheet contains the external map file location where the component characteristics are stored in tabular format. GSP component maps represent the non-linear characteristics between 2 to 5 parameters for fan, compressor, combustor and turbine. The map design values are also specified. Scaling of characteristics to the design parameters of the model component is done automatically. For all maps, parameters corrected for component entry conditions are used.

- `St.St. OD series or Transient`
  This tabsheet is either named '`St.St. OD series' or 'Transient'` depending on the case type.
  In the St.St. OD series or `Transient` tab sheet input parameters can be specified as a function of point value (St.St. OD) time (transient) in tabular format. With the `St.St. OD series/Transient input activated` checkbox the time function can be enabled or disabled during transient or steady state series calculations. When disabled or when only one row is present, the Off-design conditions are used instead (and input is constant in time). The navigator buttons are used to add, insert and delete rows, while right-clicking on a value in an existing table and selecting `Edit`, or double-clicking the value allows you to edit the value.

- `Output`
  In the `Output` tab sheet parameters which will be included in the results tables can be specified using check boxes. You are adviced to limit the number of output parameters in order to avoid very wide output tables and large amounts of output data which hinder finding the data of interest.

- `Deterioration`
  With the `Deterioration` tab sheet you can implement deterioration effects for several components. These deterioration effects apply to mass flow, pressure and efficiency.

- `Remarks`
  The `Remarks` tab sheet can be used to put down personal remarks.

For more details read Entering component data.

8.1.2.1 Component name

The component name can be specified in the top-left of GSP component data entry windows. It is used to identify components in output tables and documents and error messages. Defaults are given based on component type and the names are automatically made unique by adding numbers to the name.

8.1.2.2 Component ID string

The component ID string can be specified in the top-middle of GSP component data entry windows. If non-blank, it is used to identify component parameters such as pressure ratio PR or efficiency, which are not station number oriented (but apply to an entire component). For a compressor with ID string 'c' then compressor pressure ratio will be represented by 'PR_c' (c preceded by '_'). Component ID strings have defaults corresponding to the name of the component: c for compressor, f for fan, t for turbine, b for burner etc.
If the ID string is empty, the component number is used instead to identify component parameters. When adding components, the ID string is automatically kept unique by adding numbers to the string.

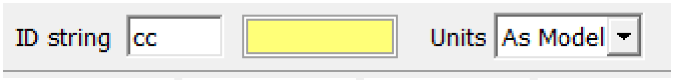
8.1.2.3 Component output data color

As of GSP 12, the output data has a customizable background color. Although the output data is already grouped into the respective component name, coloring of the background allows for even faster recognition of the component in the vast amount of output data.

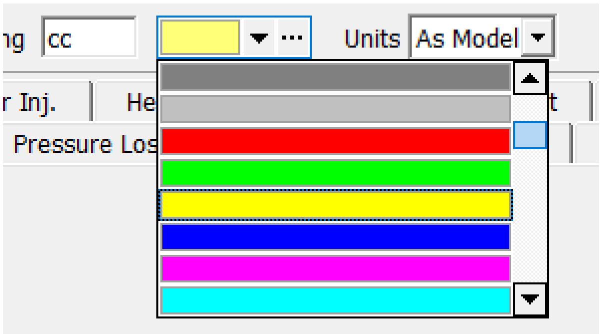
Depending on the model node, the interface is changed to be editable (in configuration nodes) or non-editable (in run case nodes). Remember to use light colors to be able to see the numbers in the output data table.

Editable:

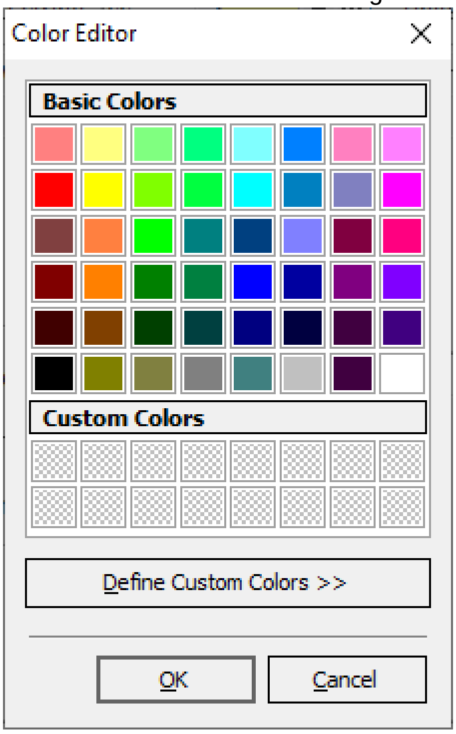
Not editable:



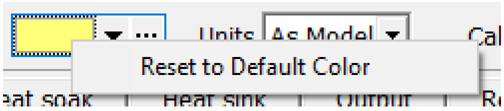
Standard color dropdown selection on clicking the down arrow next to the color:



Custom color selection when clicking the ellipsis button:



Upon right-clicking the interface a popup menu will be shown:



From the popup menu it is possible to revert the color back to the default designated design color for this component.

8.1.2.4 Component unit system

The component unit system determines the unit system for the data entry fields in a component data entry window. Choose from As Model to use the same units as selected in the General options of the active case or configuration, or alternatively choose SI or Imperial to separately choose a unit system for this component.

Note that when changing the component or model unit system, the output table should be re-created in order to avoid inconsistent units for different rows. GSP prompts the user to recreate the output tables.
Also, when changing the unit system, all input fields with common units (units defined in) covered by the unit system setting will change there values to maintain the same SI unit value that is always used internally in GSP. However, if an expression has been entered in the field instead of a valid single numerical constant, the value cannot be converted to another unit, so the expression remains unchanged and the user should adapt the expression in order to correspond to the other unit.

8.1.3 Station numbers

The station number identifies the link between two components. Usually a station number identifies a customary engine station in the gas turbine gas path. It is a user customizeable number including decimals which is per default set to the AS 755 engine station designation system. When two components are linked using GSP's drag-and-drop interface, GSP compares the station numbers of the two component links. If they do not match a dialog for specification of the number pups-up. When Cancel is pressed in the Station number dialog, the station number is set to the number that was shown initially in the dialog box.

The station numbers can be changed any time by clicking on the link icons.

The station numbers are used to identify gas path parameters such as pressures and temperatures in the output. For example: TT5 is the total temperature at station 5.

Station numbers must be unique in the model. In case the same station number on two stations is used, the output parameters will receive only the value of the last station in the gas path defined by the component order.
Do not confuse station numbers with component ID strings, which are used to identify component parameters such as pressure ratio PR for example.

Station numbers in between components or inside components (e.g. the turbine NGV station, found on the output tab sheet of the model's input window) are checked for duplicates (on initialization and on calculation).

8.1.3.1 Aerospace Standard 755

Station number designation has been standardized to unambiguously define the station interfaces. The SAE (Society of Automotive Engineers) has developed a standard (Aerospace Standard 755) for the designation of gas turbine engine station numbers Note that this naming standardization is not obligatory for use in GSP. However, it is strongly advised to conform to this standard to prevent misunderstanding in projects where multiple people access or use the engine model or the model results. Note that GSP cannot (nor the developers of GSP can) force the modeler to comply to the aerospace standard, this is left for the modeler. GSP is programmed that standard station numbering is supported as long as the modeler complies to naming the stations correctly.

The fundamental station numbers for the core stream of an engine are based on the position in the engine and the process in which the station is located.

Station number	Process boundary
0	Free stream conditions
1	Engine intake front flange, or leading edge (vehicle/engine interface)
2	Mechanical compression entrance (compressor/fan front face)
3	Last compression stage discharge or exit face (also combustor inlet face)
4	Combustor discharge (also first mechanical expansion inlet face)
5	Last mechanical expansion discharge (last turbine exit face)
6	Front face of mixer, afterburner, eduction, ejection, etc.
7	Kinetic expansion entrance (propelling nozzle inlet)
8	Kinetic expansion throat (propelling nozzle throat)
9	Kinetic expansion discharge (propelling nozzle or exhaust diffuser exit plane)

These station numbers are based on the process they take place in.

Process	Description	Between station numbers
a	Kinetic compression	0 - 2
b	Mechanical compression	2 - 3
c	Heat addition	3 - 4
d	Mechanical expansion	4 - 5
e	Mixing	5 - 7
f	Kinetic expansion	7 - 9

To comply to the standard, the station number should consist of at least 3 digits indicating the stream number process station number and a number that indicates the intermediate position in the process. Although all stations imply averaged flow properties, an additional A may be appended to the station numbering string if the station precedes a physical split of the flow path.

A valid exhaust nozzle station number would be 090, a high pressure compressor inlet would be e.g. 025, where the second stage of that compressor could be numbered 0252, etc. The exit of the intake could be 020A where the fan directly behind this intake would have an inlet core stream station number of 020 and an inlet duct stream station number of 120.

Through the output options, the user is able to define the 3-digit station string values, default, the option without cycle number prefix (the first number) is used. As GSP can optionally have more than a single cycle present in a model, an alternative method can to be used to identify the correct station number (prefixing with a 1 for the second cycle). Through the Output options it is possible to create 3-digit alternative station string which is used in the Output tables parameter output. The alternative station number strings can be shown between and onto the Component icon when enabled in the General Options.

8.1.4 Selecting components

Components on the model window can be selected by clicking on the component icon. Other already selected components on the model panel will then become unselected and subsequent operations (like editing, copying or moving) can be performed.

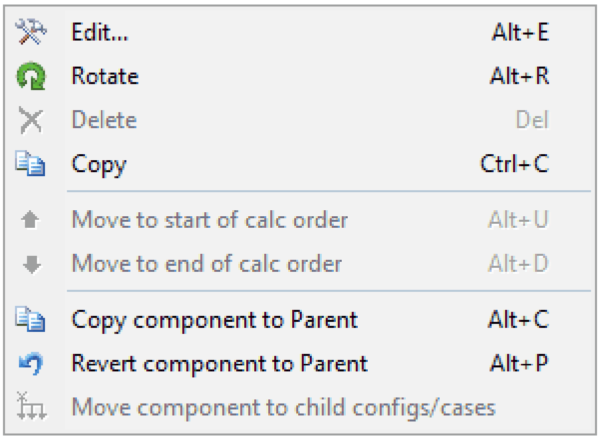
To select multiple components press the *Ctrl* or *Shift* key while clicking.

To **unselect** the components, just click on an empty area of the Model panel.

In the model project's `Edit` menu, the `Select All` (shortcut *Ctrl-A*) command can be executed to select all components.

8.1.5 Component pop-up menu

By right-clicking on a component, a pop-up menu appears with the following commands (depending on the model and the component, some options may be disabled):



All components have the ability to use:

- `Edit`
  Opens the Component Editing (data entry) form (if applicable). This action can also be performed by double-clicking the component after it has been selected.
- `Rotate`
  Rotates the component icon 90° clockwise.
- `Delete`
  Deletes the component from the model form.
- `Copy`
  Copy the component.

Control component that have the ability to be (de)activated have an extra menu option:

- `Active`
  The `Active` option toggles the active state of the control components without the need to open the component data window.

The following 2 items are disabled for standard gas path components, and enabled for control components. The calculation (and numbering; number in the top right corner of the component icon) order of control component icons is influenced by the following options. This can be very useful for advanced users to use dependent parameters in certain components (e.g. to evaluate an expression in a component that uses calculated parameters from an other component, the component number of the component evaluating the expression needs to be higher than the component calculating the needed parameters).

- `Move to start of calc order`
  Set the component number to the first control component number (usually 1)
- `Move to end of calc order`
  Set the component number to the last control component number (usually first number before first inlet number)

The following items are manipulating data in conjunction with the xml data in the tree:

- `Copy component to parent`
  This will copy all the input data fields to the parent model component. This is very useful for updating model configurations using a child design case for fine tuning your models's design/reference point.
- `Revert Component to Parent`
  Revert all component data to the data of this component in the parent configuration or case (i.e. revert all data to *inherited*). This is convenient to for example to undo changes for the component only.
- `Move component to child configs/cases`
  Move the component to child configurations and cases (child of this case or configuration). Note that this operation changed data in other cases and configurations in the project. If this operation violates case management rules (configuration components cannot be added to cases for example), the operation is aborted to avoid removing the component from child cases/configs

8.1.5.1 Editing components

By double-clicking on a component, or selecting Edit in the pop-up menu, the component data window appears, allowing the specification of the component model data.

Shortcut is `Alt + E` or `Enter`.

8.1.5.2 Rotating components

Rotate the component icon if this makes linking with other components easier. For complex models (for example with recuperator heat exchangers) rotating the icons is necessary to enhance model surveyability. The Link bar component often has to be rotated 90 degrees to enable vertical component links.

Shortcut is `Alt + R`

8.1.5.3 Activate control components

This option toggles the active state of the control components without the need to open the component data window.

Shortcut is `Alt + A`

8.1.6 Copying, moving and deleting components

Apart from copying components from the libraries when building new models, components can be moved and copied on the model panels and to and from other model panels. This allows copying components including specific component data from other engine models, enabling the user to create model projects serving as libraries similar to component libraries with predefined template components.

On a model panel, the user can move a selected component by just dragging it across the white area of the panel.

As with normal Windows operation, pressing the *Ctrl* button while dragging makes a copy of the component. With the *Ctrl* key pressed, components can also be copied from other model panels.

Also, the (Windows-) customary copy, cut and paste commands can be used (available in the `Edit` menu on the model panel and as keyboard shortcuts (*Ctrl-C, Ctrl-X* and *Ctrl-V*), which work with an invisible "component clipboard" for copying and moving operations.

When multiple components are selected, these operations apply to all selected components. GSP automatically repositions components on the form to prevent invisible components located outside the forms. Use the scroll bars to scroll the model panel if necessary or resize the model panel. In case component overlap results after a copy or move action, the modeler will be visibly notified.

The Delete command in the Edit menu, pop-up menu or invoked by pressing the Delete key, removes the selected components. Not all components can be deleted, the option may therefore be disabled. Note that deleting components, unless there is no parent component, just moves the component to the deleted list where it is ignored by the program. There is an option to restore the component through the Deprecated components window (double-click entry in the list).

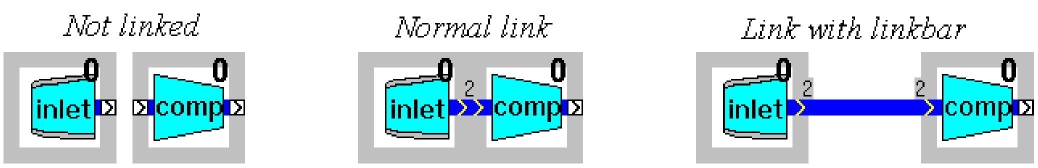
Since the case management features are incorporated in the modeling environment, additional constraints are set on moving and deleting components. Note that a component of a Configuration or a Case can have an ancestor, deleting a child must be inhibited to ensure a proper model structure since this structure is based on inheritance.

Moving of Configuration type components is inhibited in a Case model, since this results in a change of the configuration of the engine layout. Note that the Case model is an instance of the Configuration with a sole existence to run the configuration as is. In order to provide the case simulation of input, specific case control components can be used alongside the normal (numerical) data input fields.

8.1.7 Linking components

When building new models, components have to be arranged in a particular configuration in order to represent the actual gas turbine engine model in a model window. GSP uses *linked objects* represented by link icons to facilitate interaction among the components. The link elements are visualized by small rectangles on the components icons with double chevrons indicating the direction of data.

The user must position the components in a manner that the links touch each other. When a link can be established, the rectangles turn blue for gas path links, and black for control links. When a certain distance between components is desired, a Link Bar may be used to establish the link over a larger distance. With the component pop-up menu, component icons and link bars can be rotated.



8.1.8 Component model fidelity

GSP component models may have different levens of fidelity. GSP's standard component models are all non-dimensional. Some custom component models are 1-dimensional. Non-dimensional gas turbine component models use gas properties, averaged over the flow cross areas at entry and exit stations only. For example, combustor discharge gas properties are represesented with single pressures, temperatures, flow rates, Mach numbers, densities, fuel air ratio's etc. Pattern effects are not taken into account. For general (i.e. "whole system") gas turbine performance analysis, this assumption does not produce large inaccuracies.

Note that GSP component models do not need to be limited to non-dimensional or even 1-dimensional. Due to GSP's flexibility there is principally no limitation to modelling fidelity inside a component model. In some engine specific custom component libraries, dimensional models exist, such as the 1-dimensional combustor model which calculates combustion kinetics depending on some combustor geometrical properties.

8.1.9 Component libraries introduction

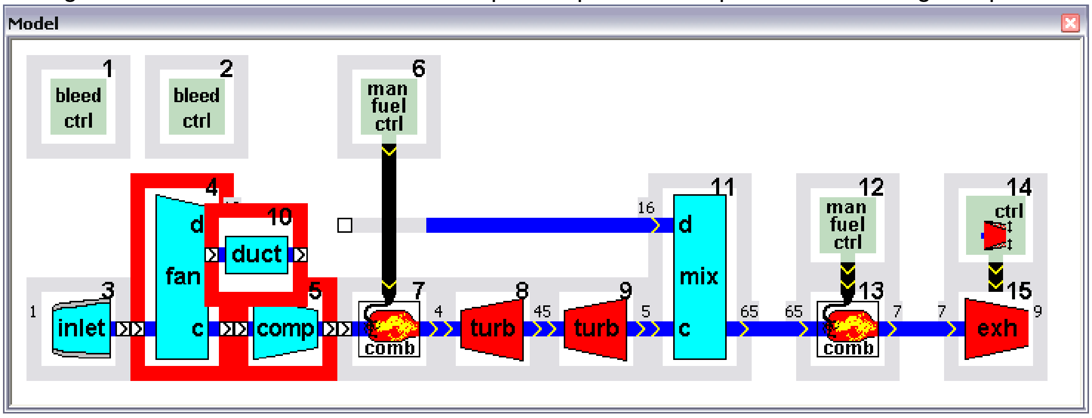
Component libraries are represented by component libraries and contain collections of generic components that can be copied to project windows in order to build gas turbine models.

NLR offers several application specific custom component libraries with component models dedicated to certain gas turbine engine systems. Please contact NLR regarding custom features in components.

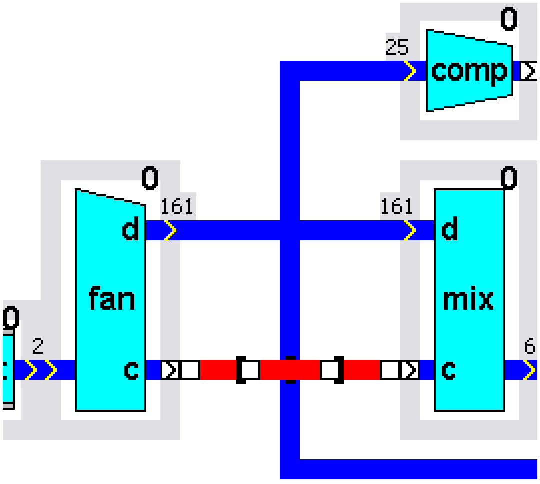
8.1.10 Component Overlap

To prevent modelers from noticing overlaying components, which can happen when in a configuration or case model high up in the project tree a new component is introduced, the border of the component icon (of both the overlaying and underlaying components) will be colored red.

The figure below demonstrates a duct component placed on top of some existing components:



For overlying link (bar) components, the link bars are only colored red if the orientation is the same. This means that link bar crossings will not be regarded as overlapped components. In the figure below a short horizontal link bar is placed over the lower fan to mixer link bar and crosses the link bar to the compressor component:



8.1.11 Component design point

For most components, design point data needs to be specified to define the component operating point corresponding to the design gas turbine system operating point. Please note that "design point" as used in GSP might cause confusion, the design point of the model does not necessarily need to be an actual design point of the gas turbine in question. It is actually better to refer to the model design point as the "cycle reference point". The model reference point is usually a high power operating point where the modeler has the most avaiable data from literature, spec./fact sheets or measurement data. Usually the cruise or take off points are used for this.

Depending on the component, a number of component or design parameters must be specified in the `Design` tab sheet in the component data specification window. For some components also other tab sheets hold design point data such as the map design point in the `Map` tab sheet of the compressor and turbine components. Design point data are indicated by a **navy blue** color. Furthermore, since the introduction of GSP 11, design data can only be changed in configuration models or in dedicated design run case models.

8.1.12 Component output parameters

In the `Output` tab sheet of the component data editing window, the parameters that will be included in the results tables can be specified using check boxes. It is adviced to limit the number of output parameters in order to avoid very wide output tables and large amounts of output data which hinder finding the data of interest.

Note that the selection of output data also represents the parameters that can be selected to be plotted in the graphical output.

8.1.13 Deprecated components

To ensure backwards compatibility, we maintain components that can be replaced by other, more advanced, components. A message will be shown to remind the modeler that a component has been deprecated. A general option has been added to ignore this warning message in General options.

Examples of deprecated components are Thrust Control, Rotor Speed Control and EPR Control components. The components cannot be controlled by components from the Case Control Component library. Their functionality is replaced by the equation schedule components. The components are merely kept for backward compatibility, or for simple control.

8.2 Component off-design effects

When performing off-design simulations several effects are present depending on model setup:

- Dynamic effects
  - Rotor inertia effects
  - Heat soakage effects
  - Volume effects
- Pressure loss models
- Variable geometry
- Deterioration
- Component Maps
- Heat sink (heat transfer) effects

8.2.1 Variable geometry

A number of components include models for variable geometry effects such as variable exhaust nozzles (VEN's) and variable compressor bleed valves (VBV's) and variable stator vanes (VSV's) in compressors. Variable stator vane schedules can be specified as tabular "map modifier" functions of corrected rotor speed in the compressor's `Variable geometry` tab sheet.

8.2.2 Deterioration

A number of components include `Deterioration` tabsheets for specification of map modifiers representing the effect of deterioration. Refer to literature for quantification of deterioration effects. A good way to analyse deterioration effects is to use steady state series calculations for parameter sweeps with varying deterioration map modifiers.

A typical turbine deterioration case can for example be represented by a 4% decrease in turbine efficiency combined with a 2% increase in massflow. A typical compressor deterioration case (fouling) can for example be represented by a 2% decrease in compressor efficiency combined with a 2% decrease in massflow.

8.2.3 Component maps

GSP applies tabular component characteristic maps to determine the relations among up to 5 component operating point parameters. The maps are stored in separate files, or alternatively can be stored embedded, and are compatible with the GasTurb [1] format. The operating point parameters are corrected for standard conditions in order to make the characteristic relations independent of the component inlet conditions. One or more of the following 5 parameters generally are used for component characteristics :

- corrected mass flow
- corrected rotational speed
- pressure ratio
- component efficiency
- Reynolds index
  The Reynolds index is generally omitted since it only has a relatively small effect on

component performance. However, *Reynolds correction factors* may be specified in the map table based on the *Reynolds index*.

Pressure ratio (or relative pressure loss) and corrected entry mass flow are often used as map parameters. Corrected rotational speed and isentropic efficiency only are related to turbo machinery (i.e. compressors, fans and turbines). Up to three *map input parameters* (including the Reynolds index) are used to define the map operating point and determine the other *map output parameter* values.

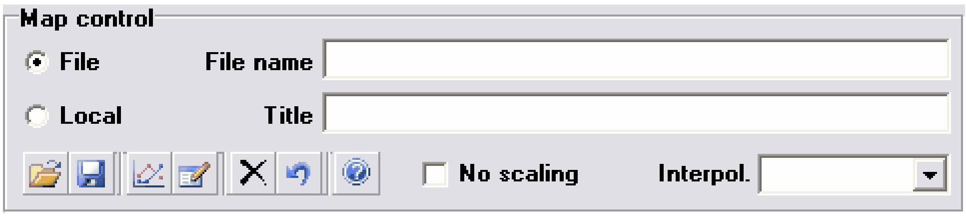
Tabular characteristics can be scaled in order to be able to use them for different gasturbines. If the parameter values of the map design point do not match those in the design point of the gasturbine to be modeled, scaling factors are determined during the design point calculation.

Corrected or reduced parameters are often used for representation of component characteristics and overall engine performance. The off-design component maps often use corrected parameter groups to represent corrected mass flow and corrected or referred speed. Full correction (dimensionless) includes the diameter of the gas turbine and the gas properties. When a single gas turbine is considered, the diameter is not important. The gas properties are important when the real gas effects are considered since gamma is dependent on the temperature and R is dependent on the gas composition.

Note: The default GSP installation copies all model and map (.map) files to the GSP projects directory. Map filenames paths are relative to the GSP directory. Therefore, when copying models or project files, with the map location set to file, and maps from one computer to another, an error message can appear due to a change in path.*

8.2.3.1 Map Handling

A generic map control interface is used to control maps in various components. Since not all GUI items are useful for specific implementations some (GUI) elements may be disabled or made invisible. The following picture shows the generic layout of the map control.



Typical map control GUI layout

- Map type
  - `File`
    A map is specified by a link where it can be found. In distributing models the maps should be given as well, and should be placed in exactly the same location as specified by the `File name` input string.
  - `Local`
    The map will be stored internally inside the project file. File stored

- `File name`
  Path and filename to the loaded map. If the map resides next to the model/project, no path will be displayed. Manual file editing is disabled, selecting maps must be done using the browse button.
  The color of the file name informs the user of the existence/state of the map file/object.

- **Black** font
  A black font is used for a valid map location, i.e. a file with the specified name is found, for the `File` map option.
- **Bright blue** italic font
  Embedding a map will immediately result in a bright blue font, this means that the map object has been embedded in the project node, and the path to the file still exists.
- **Red** italic font
  For embedded maps the red font is introduced to indicate that the map path originally used to obtain the map data (on embedding map) is not valid anymore. An invalid map location can be the result of moving the project file to a different location/directory, or when the map files are moved.

- `Title`
  Optional map title. If a map file has a title specified, the map title will be display after loading the map.

- Note that controlling maps is only possible using the buttons in the lower left corner. From left to right these buttons stand for:
  - `Open map file (CTRL + O)`
    A map dialog will be opened to select a map file
  - `Save to file (CTRL + S)`
    Save the current loaded map to file
  - `Show map graph`
    Display the map's graphical representation
  - `Edit/view data (CTRL + E)`
    This will open the current loaded map in the default text editor
  - `Clear map data (CTRL + X)`
    This action will erase/clear the current loaded map
  - `Revert to model data as loaded when entering this config/case`
    Revert the map to the state the model config/case was entered (or last config/case save point)
  - `Help (F1)`
    Opens help

- `No scaling`
  Resets map scale factors off-design to 1. This disables the automatic scaling to design point data for off-design analysis (no map scaling). This feature can be used when the component characteristics are known from testing or analysis. When enabled (of the shelf) component matching is possible.

- `Interpol.`
  This drop-down box determines the type of interpolation. `Linear` interpolation is the default, but can be changed to `Quadratic`.

Note that some specific elements have been invisible when they are not applicable to the map (e.g. `No scaling` is sometimes not used).

8.2.3.2 Map graph

The most convenient way to represent component performance characteristics ('maps') is in a graph. Several types of maps exists for different component types. Some component types like compressors, fans combustors and turbines have special graphical representation layouts that are commonly used in the gas turbine community. For many other relations such as in scheduler-, control- and case control components, simple 1-D and 2-D maps are used for which simple X-Y plots are used for graphical representation.
In maps that are scaled, a yellow rectangle represents the map design point (DP) to which the engine design point is scaled. If the map graph is scaled using the `Map | Scale to model design point` menu option, then naturally the rectangle represents the actual engine DP. See example of a compressor map. Map graphs can be saved or copied to the clipboard or to files in bitmap or meta file format using the `Copy` and `File` menus in the graph window. Note that when printing

or exporting the map graph to bitmap or meta formats, only if the map is scaled also the rectangle is drawn in the map.

8.2.3.3 Component map scaling

Tabular characteristics can be scaled in order to be able to use them for different gas turbines. If the parameter values of the map design point do not match those in the design point of the gas turbine to be modeled, scaling factors are determined during the design point calculation. These scaling factors are used to convert the input parameters to the map scale before the map data is accessed, and then to convert the map output parameters back to the gas turbine scale. For example, a compressor map with a nominal 100 kg/s mass flow and a pressure ratio PR of 5 can be used to represent a slightly larger but geometrically similar compressor with higher pressure ratio (e.g. 110 kg/s and PR = 6 results in scaling factors 1.1 and (6-1)/(5-1)=5/4 respectively). Scale factors for map output parameters such as isentropic efficiency are determined from the ratio of the user specified component design value (e.g. design efficiency) and the map efficiency in the map design point.
For small differences (<25%), scaling usually does not add large errors (i.e. deviations from actual component performance). Scaling for larger difference will introduce large error margins due to the changing Reynolds number.

Linear interpolation is used to obtain map parameter values from the tables. This requires attention to the resolution of the table parameter step size, especially with very non-linear relations. The SmoothC and SmoothT [6, 7] , and GasTurb [2] programs can be used to quickly increase map table resolution.

Note: The default GSP installation copies all model and map (.map) files to the GSP projects directory. Map filenames paths are relative to the GSP directory. Therefore, when copying models and maps from one computer to another, an error may appear due to a change in path.*

8.2.3.4 Beta parameter

In the maps for compressor, turbine and derived components (e.g. the fan), the *beta* parameter is used to avoid numerical convergence problems during iterations towards the operating point solutions. With the typical relations between corrected mass flow and pressure ratio for constant rotational speed for these turbo-machinery components, either one of the variables can become independent of the other. For example, the constant speed curve in a compressor map can be nearly horizontal or nearly vertical. This causes numerical problems, since for one rotor speed and pressure ratio, multiple values for mass flow are possible. To avoid these numerical problems, the beta parameter is added, representing a relation between pressure ratio and corrected mass flow for which constant beta lines or curves are virtually perpendicular to the constant corrected speed curves in the map graph. Beta values are equidistant, ranging between 0 and 1. The beta parameter is also used in GasTurb [1].

8.2.3.5 Corrected speed

Below, a derivation for the normalized corrected spool speed is shown. This demonstrates the difference between real gas corrections and ideal gas corrections.

The corrected spool speed is defined as (including real gas effects):

$$N_c = \frac{\frac{N\,DI}{\sqrt{\gamma R T}}}{\frac{N_{ref} DI_{ref}}{\sqrt{(\gamma R T)_{ref}}}} = \frac{N}{N_{ref}} \frac{DI\sqrt{(\gamma R T)_{ref}}}{DI_{ref}\sqrt{\gamma R T}} = \frac{\frac{N}{N_{ref}}\frac{DI}{DI_{ref}}}{\frac{\sqrt{\gamma R T}}{\sqrt{(\gamma_{ref} R_{ref} T_{ref})}}}$$

The diameter DI is not of interest if the same machine is considered, the equation can be rewritten to (including real gas effects):

$$N_c = \frac{\frac{N}{N_{ref}}}{\frac{\sqrt{\gamma R T}}{\sqrt{(\gamma_{ref} R_{ref} T_{ref})}}} = \frac{\frac{N}{N_{ref}}}{\sqrt{\frac{T}{T_{ref}}}\sqrt{\frac{\gamma R}{\gamma_{ref} R_{ref}}}}$$

This equation reduces to:

$$N_c = \frac{\frac{N}{N_{ref}}}{\sqrt{\frac{T}{T_{ref}}}} = \frac{\frac{N}{N_{ref}}}{\sqrt{\theta}}$$

if no real gas effects are considered. Note that the corrected spool speeds used in the component characteristic maps for compressors and turbines in GSP use this relation. Without a reference spool speed the relation becomes:

$$N_c = \frac{N}{\sqrt{\theta}}$$

8.2.3.6 Corrected flow

Below, a derivation for the normalized corrected mass flow is shown. This demonstrates the difference between real gas corrections and ideal gas corrections.

For the corrected mass flow a similar derivation exists:

$$W_c = \frac{\frac{W\sqrt{TR}}{DI^2 P\sqrt{\gamma}}}{\frac{W_{ref}\sqrt{(TR)_{ref}}}{DI_{ref}^2 P_{ref}\sqrt{\gamma_{ref}}}} = \frac{W\sqrt{TR}}{W_{ref}\sqrt{(TR)_{ref}}} \frac{DI_{ref}^2 P_{ref}\sqrt{\gamma_{ref}}}{DI^2 P\sqrt{\gamma}}$$

The diameter DI is not of interest if the same machine is considered, and Wref is usually 1, the equation can be rewritten to (including real gas effects):

$$W_c = \frac{\frac{W\sqrt{TR}}{P\sqrt{\gamma}}}{\frac{W_{ref}\sqrt{(TR)_{ref}}}{P_{ref}\sqrt{\gamma_{ref}}}} = \frac{W\sqrt{TR}}{\sqrt{(TR)_{ref}}} \frac{P_{ref}\sqrt{\gamma_{ref}}}{P\sqrt{\gamma}} = \frac{W\sqrt{\frac{T}{T_{ref}}}}{\frac{P}{P_{ref}}} \frac{\sqrt{R\,\gamma_{ref}}}{\sqrt{R_{ref}\,\gamma}}$$

This reduces to:

$$W_c = \frac{W\sqrt{\dfrac{T}{T_{ref}}}}{\dfrac{P}{P_{ref}}} = \frac{W\sqrt{\theta}}{\delta}$$

if no real gas effects are considered.

The Greek characters theta and delta represent the correction factors to refer engine parameters to ISA sea level static conditions.

8.2.3.7 Map formats

GSP applies tabular component characteristic maps to determine the relations among up to 5 component operating point parameters. The maps are stored in separate files, or alternatively can be stored embedded, and are compatible with the MTU / GasTurb [1] format. Map files are installed upon installation of GSP in the GSP sample models folder.

Note that in the map formats described hereafter, text with a blue font is information and therefore not part of the actual map file.

The different map formats that are used in GSP are briefly described hereafter. The header refers to the application of the map, while in brackets the amount of input and output parameters are given. A compressor map is defined by 2 input parameters (Beta parameter and Speed parameter), to find Efficiency, Pressure Ratio and Corrected Mass Flow, hence 2 in 3 out.

Example maps can be found in the default sample projects folder or embedded in the supplied sample project files.

8.2.3.7.1 Map table format

A table is a set of numbers and contains argument values (A), parameter values (P) and function values (F):

Key	A[1]	A[2]	A[3]	A[4]
P[1]	F[1,1]	F[1,2]	F[1,3]	F[1,4]
P[2]	F[2,1]	F[2,2]	F[2,3]	F[2,4]
P[3]	F[3,1]	F[3,2]	F[3,3]	F[3,4]

The first number of a table is the **table key** which is composed from the number of rows and columns of the table:
Key = number of rows + number of columns/1000

The number of rows is one more than the number of parameter values in the table and the number of columns is one more than the number of arguments in the table. The key for the table above would be 4.005, for example.

A table starts always on a new line and begins with the key. After the key follow the first four argument values, separated by at least one "blank" from each other. The rest of the argument

values are on additional lines (five numbers per line). Only on the last line with argument values there may be less than five values (columns).
Parameter values must always begin a new line and the first four function values follow on the same line. The rest of the function values are arranged as described for the argument values.
The data need not be in specific column positions, but there must be **at least one blank between each number**.

NOTE!
Limits are present on the amount of columns and rows a map table may consist of:

- Maximum amount of columns (argument values!)
  Currently a maximum of **40** argument values are supported (remember the limit per line in map file is 5 values, the parameter value not included, implying that 8 lines can be used at the most).
- Maximum amount of rows (parameter values!)
  Currently a maximum of **150** parameter values are supported.

8.2.3.7.2 Compressor/Fan map format

Compressor/fan map format (2 in 3 out)

A compressor/fan map is described by a set of four tables stored in one file. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=x1 f = y1 RNI = x2 f = y2
Table Content Keyword
↓
Mass Flow
<Table with ß=Argument, Speed=Parameter, Mass Flow=Function Value>
Efficiency
<Table with ß=Argument, Speed=Parameter, Efficiency=Function Value>
Pressure Ratio
<Table with ß=Argument, Speed=Parameter, Pressure Ratio=Function Value>
Surge Line
<Table with Surge Point Mass Flow = Argument, 1.0 as Parameter, Surge Point Pressure Ratio=Function Value>
```

8.2.3.7.3 Turbine map format

Turbine map format (2 in 3 out)

A turbine map is described by a set of four tables stored in one file. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=x1 f = y1 RNI = x2 f = y2
Table Content Keyword
↓
Min Pressure Ratio
<Table with Speed=Argument, Min Pressure Ratio=Function Value>
Max Pressure Ratio
<Table with Speed=Argument, Max Pressure Ratio=Function Value>
Mass Flow
<Table with ß=Argument, Speed=Parameter, Mass Flow=Function Value>
Efficiency
<Table with ß=Argument, Speed=Parameter, Efficiency=Function Value>
```

8.2.3.7.4 Duct map format

Duct Map (1 in 1 out)

A duct map is described by a single table containing 2 columns, Wc and dPrel

```
Map Type Indicator Map Title
↓                  ↓
99                 Example Pressure loss Map
Reynolds correction data
↓
Reynolds: RNI=0.100 f=0.950 RNI=1.000 f=1.000
Table Content Keyword
↓
Wc - dPrel
Table data
↓
0.000E+00  0.000E+00
1.250E+01  3.000E-02
1.350E+01  1.000E-01
etc.
```

8.2.3.7.5 Inlet map format

Inlet map (2 in 1 out)

A duct map is described by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Ram Recovery Factor
<Table with Mach=Argument, Corrected inlet exit mass flow=Parameter, Ram Recovery factor=Function Value>

Note that the mass flow should be increasing from top to bottom!
```

8.2.3.7.6 Combustor map format

Combustor map (2 in 1 out)

A duct map is described by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Combustor Efficiency
<Table with delta T combustor=Argument, Delta (Pt/Pref @ Sea Level Standard)=Parameter, Combustor
efficiency=Function Value>
```

8.2.3.7.7 Heat exchanger effectiveness map format

Heat exchanger effectiveness map (2 in 1 out)

A duct map is described by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Effectiveness
<Table with Heat capacity rate flow 1 (W*Cp)=Argument, Heat capacity rate flow 2 (W*Cp)=Parameter,
Effectiveness=Function Value>
```

8.2.3.7.8 Afterburner maps format

Afterburner (3 maps of 1 in 1 out)

For an afterburner a set of 3 maps is used to determine the combustion efficiency.

```
Map Type Indicator Map Title
↓                  ↓
99                 F100PW200 Afterburner combustion efficiency map / as function of relative FAR
Reynolds correction data
↓
Reynolds: RNI=0.100 f=0.950 RNI=1.000 f=1.000
Table Content Keyword
↓
FARrel - ETArel
Table data
↓
0.0390   0.9400
0.0585   0.9887
0.0732   1.0193
etc.

Map Type Indicator Map Title
↓                  ↓
99                 F100PW200 Afterburner combustion efficiency map / as function of relative FAR
Reynolds correction data
↓
Reynolds: RNI=0.100 f=1.000 RNI=1.000 f=1.000
Table Content Keyword
↓
FARrel - ETArel
Table data
↓
0.0390   0.9400
0.0585   0.9887
0.0732   1.0193
etc.

Map Type Indicator Map Title
↓                  ↓
99                 Afterburner combustion efficiency Mach Correction map
Reynolds correction data
↓
Reynolds: RNI=0.100 f=1.000 RNI=1.000 f=1.000
Table Content Keyword
↓
dMachrel - dETArel
Table data
↓
1.000   -0.000
1.071   -0.013
1.190   -0.041
etc.

Map Type Indicator Map Title
↓                  ↓
99                 Afterburner combustion efficiency Pressure Correction map
Reynolds correction data
↓
Reynolds: RNI=0.100 f=1.000 RNI=1.000 f=1.000
Table Content Keyword
↓
dPrel - dETArel
Table data
↓
0.220    -0.1420
0.2267   -0.1250
0.2500   -0.1000
etc.
```

8.2.3.7.9 Propeller map format

Propeller map (2 in 3 out)

A propeller map is described by a set of four tables stored in one file. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text
Reynolds correction data
↓
Reynolds: RNI=x1 f = y1 RNI = x2 f = y2
Table Content Keyword
↓
Advance Ratio
<Table with ß=Argument, Propeller angle (Beta)=Parameter, Advance Ratio=Function Value>
Efficiency
<Table with ß=Argument, Propeller angle (Beta)=Parameter, Efficiency=Function Value>
Power Coefficient
<Table with ß=Argument, Propeller angle (Beta)=Parameter, Power Coefficient=Function Value>
Static Performance Cf/Cpw = f(Cpw)
<Table with Power Coefficient=Argument, 1.0 as Parameter, Static Performance Cf/Cpw = f(Cpw)=Function
Value>
```

8.2.3.7.10 Schedules map format

Schedules (2 in 1 out)

A schedule (2 in 1 out) is described by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text(e.g. Bleed flow as function of Mach and Altitude)
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Schedule
<Table with Parameter1(e.g. Mach)=Argument, Parameter2(e.g. Altitude)=Parameter, Map Result (Bleed flow)
=Function Value>

Note that both the Argument and the Parameter should be increasing from respectively left to right and top
to bottom!
```

8.2.3.7.11 Installation loss maps format

Spill drag map

A spill drag map is used to define the spill drag of the intake and is defined by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text (e.g. Spill Coefficients as function of Mach and Capture Ratio)
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Spillcoeffs
  4.00800    0.0    0.2    0.4    0.6
      0.8    1.0    2.0
  0.00000   1.00   0.64   0.36   0.16
     0.04   0.00   0.00
  1.00000   1.28   0.88   0.58   0.31
     0.13   0.00   0.00
  2.00000   1.55   1.20   0.87   0.58
     0.28   0.00   0.00
<Table with Mach=Argument, Capture Ratio=Parameter, Spill drag coefficient(Cd_spill)=Function Value>
```

Afterbody drag map

An afterbody drag map is used to define the drag of the exit and is defined by a single table. The format of a table is described in Map table format.

```
Map Type Indicator Map Title
↓                  ↓
99                 Arbitrary Text (e.g. AftBody Coeffs as func. of Mach and NPR)
Reynolds correction data
↓
Reynolds: RNI=0.1 f=1 RNI=1 f=1
Table Content Keyword
↓
Aftbodycoeffs
<Table with Mach=Argument, Nozzle Pressure Ratio=Parameter, Afterbody drag coefficient(Cd_aft)=Function
Value>
```

8.3 Component Models

The GSP components which in a particular arrangement represent a gas turbine model, simulate component performance using algorithms calculating changes in gas properties, mass flows, turbomachinery shaft power, control system logic etc. The components internals describe the mathematical description of the gas turbine cycle/arrangement. This mathematical description includes the definition of states and error equations to build the complete set of non-linear differential equations. The set of equations can be directly changed by the component model options.

The following main component model categories can be identified:

- Gas path components
- Link bar component
- Linkable components

- Unlinked components
- Control components

The following items represent sub-components within the component models, necessary to enable the transmission of power through drive shafts and compressor bleed flow to other components (such as turbines for blade cooling):
- Shaft objects
- Bleedflow objects

Due to the advanced software development principles applied in GSP 12, new or specific application dedicated components can be derived from existing ones very rapidly.

8.3.1 Component states and errors

GSP internally represents a gas turbine engine system model as a 'virtual' set of non-linear differential equations. GSP's generic solver solves these equations in order to find a valid steady state or transient operating point (for more information see the Technical Manual and Reference [4].

The number of states and/or errors is affected by the Component model options for the following components:

- Fan
- Compressor
- Turbine
- Manual fuel control
- Gas generator fuel control,
- Turboshaft fuel control

Failure of correct setting of the Component model options affecting the states and error configuration of the model will be reported as either an error due to the number of model states and (equation) errors not being equal or an error detecting more than one component trying to (user) specify the same shaft.

8.3.2 Component model options

The component `Model Options` in the General tab sheet determine how the component model is used in relation to the whole engine system model. The options depend on model type but in many cases determine what component performance variables are used as component states and/or error variables in the differential equations.
Often, the setting of an option in one component requires the a specific specific setting in another.
The fuel flow in the Manual Fuel control component can be specified as a 'free state' (instead as a user specified input parameter) in order to calculate an off-design operating point with a user specified gas generator rotor speed. This requires A compressor component model option to be set as "externally controlled rotor speed' and the 'Power balance at rotor speed' option for the gas generator turbine component.

8.3.3 Gas path components

Gas path components are the main items necessary to build a gas turbine model having one property in common: they all conduct an air or gas flow. There always is at least one entry and one exit, enabling the receiving or passing of flow from/to other components. Usually, the properties of the gas flowing though are affected in some way and some components (like turbomachinery or heat exchangers) also require or deliver power or heat flow. Gas path components are available in the Gas Path Components.

8.3.4 Linkable components

Linkable components are components with one or more link icons, meant to establish a link with another component (such as a link for a gas flow from one into the other component). Linkable components are the opposite of unlinked components.

8.3.5 Unlinked components

Unlinked components are components without link icons. The are not meant to establish a link with another component (such as a link for a gas flow from one into the other component) and are used to control global model variables like bleed flows or power turbine loads. Unlinked components are the opposite of linked components.

8.3.6 Control components

Control components are used to model the engine systems which actively affect the engine operating point.
GSP offers six standard controller types in the control component library in the following categories:

Fuel control components

These components control the fuel flow in order to control power level or maintain rotor speeds, and include:

- Manual Fuel Flow Control
  Direct manual control of the fuel flow rate.
- Gas Generator Fuel Control
  A generic control system component for modelling shaft speed governor controls (Proportional-Integral-Differential PID control) with acceleration control included. Use this control for simple turbojet and turbofan engine control.
- Turboshaft Fuel Control
  A component inheriting all characteristics of the above governor control with in addition a separate generic power turbine shaft speed control PID governor model. Use this control for customary turboshaft engine controls.

Variable geometry components

- Compressor Bleed Control
  Controls off-design bleed flow rates.
- Manual Variable Exhaust Nozzle Control
  Off-design manual control of a variable exhaust nozzle

Variable inlet guide vanes (VIGV) and stator vane (VSV) control components are inherited from these components and obtainable as custom components (not included in the public GSP version).

Load control components

These components provide a means to specify varying off-design (transient) power turbine loads in terms of torque and/or power levels, and include:

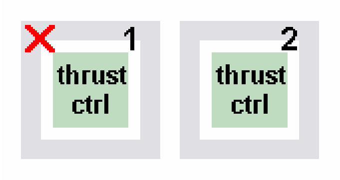
- Shaft Load control

Propellor and helicopter rotor models calculating turbine loads are inherited from this component and obtainable as custom component (not included in the public GSP version).

Note: steady state and steady state series calculations may become problematic when control system components with engine parameter feedback depending on time (such as Gas Generator Fuel control and Turboshaft Fuel control components) are used in the model. The best way to calculate a steady state point then is to stabilize at the particular operating point using a transient calculation first before performing a steady state calculation.

Most control components can be enabled/disabled by checking/unchecking the `active` checkbox on the `general tab sheet` of the component property window. A disabled component will display a red cross in the top left corner of the component icon in the model window.

Below displays a disabled (thrust controller nr.1) and an enabled (thrust controller nr.2) controller



8.4 Component Libraries

The library windows contain the component libraries with the components necessary to build GSP gas turbine engine models. The most frequently used library will likely be the standard library. Components on library windows are fixed "templates" that cannot be modified. However, the user can make a default project containing unlinked user specified components to represent a user customizable library.

Registered custom GSP versions may include extra component libraries with advanced or company specific components or libraries.

The following component categories are available:

- Gas Path Components
- Engine Control Components
- Case Control Components
- Scheduling Components
- Auxiliary Components
- Miscellaneous Components
- Custom Components

8.4.1 Gas Path Components

Gas path components represent elements in the model of the gas turbine process and have up to 2 inlets and/or 2 exits.

Gas path components have (in most cases) the following modeling elements in common:
- Heat soakage effects (dynamic thermal effects on transient performance), specified in the `Heat soakage` tab sheet.
- Heat sink effects (steady-state and transient heat transfer effects among components and/or with ambient), specified in the `Heat sink` tab sheet. The gas path component can be

connected to one or more heat sink components in the heat sink table. Each row in the table represents a connection with a heat sink with separate user specified heat transfer model data.

- Volume effects (dynamic effects on transient performance)

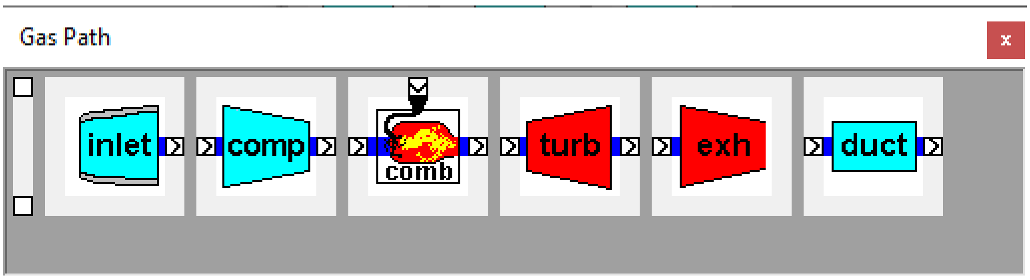
The gas path components are included in the following libraries:

- Gas Path Component Library
  Contains the standard single input link and/or output(exit) link gas path components.
- Multi in/out Component Library
  Contains multiple input link and/or output link gas path components (e.g. flow splitters/joiners, multi flow etc.).
- Special Gas Path Component Library
  Contains specific custom gas path components.

8.4.1.1 Gas Path Component Library

The GSP Gas Path component library includes the following components:

- Link-bar
- Inlet
- Compressor
- Combustor (also Afterburner)
- Turbine
- Exhaust nozzle
- Duct



Gas path library panel

Gas path component models share a number of common properties and model elements.

8.4.1.1.1 Common Gas Path component elements

GSP gas path component models are the primary building blocks for gas turbine models. Most of the gas path component represent a stage in the gas turbine thermodynamic cycle, such as Inlet, Compressor, Combustor and Turbine subsequently for a simple cycle single spool gas turbine. With these components stacked together, only the addition of a Manual fuel controller is then required to complete the model for a simple jet engine like in the TJET.MXL project used in the quick start basics tutorial.
In all gas path component models, gas is entering one or more inlets, then is subjected to some process affecting the gas conditions and/or composition before exiting through one or more exits. The component inlets and exits usually correspond to engine stations, where specific gas conditions and/or compositions are specified or calculated for the primary gas flow of the cycle. Gas path component models share the following elements:

- Primary gas flows enter the component via 1 or 2 component inlets.
- Primary gas flows leave the component via 1 or 2 component exits.

- Output parameters written to the output tables are defined through the `Output` tab sheet of the component
- Secondary gas flows can enter or exit the component depending on the component model type.
- Gas condition and composition is calculated for entry and exit stations. Only in special cases intermediate stations or locations exist where gas conditions and composition is calculated.
- In most cases (and per default) only total temperature and pressure is required for the cycle calculation since enthalpy is used for the conservation of energy equations.
- Static conditions (static temperature, static pressure, mach, velocity) can be calculated in most gas path components using the `Exit static conditions` box in the gas path component data entry window `Design` tab sheet.
- Heat transfer can be modeled 2 ways:
  - o Using the `Heat sink` tab sheet to connect to a Heat sink component for both steady-state and transient heat transfer to other components or the environment
  - o Heat soakage heat transfer using the Heat Soak tab sheet for heat transfer during transient only heating up or cooling down the gas path surrounding walls.
- Volume effects using the `Vol.dynamics` tab sheet for representing volume effects during transients.

8.4.1.1.1.1 Static conditions

Static conditions can be calculated at every engine station using the using the `Exit static conditions` box in (most) gas path components design tab sheets.

With the total properties (temperature and pressure) and flow rate calculated at every cycle calculation iteration step for evaluation the model equations for conservation of mass and energy, static pressure and temperature can only be calculated if flow velocity is know. Flow velocity can only be calculated if flow cross area A is given for the particular station. Alternatively, also velocity or mach number can be specified and then flow area calculated. In summary, with a value specified for either area, velocity or mach number, the other 2 can be calculated. Iteration is required to accurately maintain the correct relations for total to static conditions including the depending of density and sonic velocity of static temperature and pressure.

In GSP, engine station static conditions are calculated in the component just upstream of the station which means the component exit static condition is calculation. Since this corresponds to the inlet static condition of the next component, separate inlet static condition calculation is omitted to prevent duplication and conflicting data. So if a component inlet static condition is desired, use the upstream component the `Exit static conditions` option instead. Select the desired parameter for specifying static conditions (area, mach number or velocity) and enter an appropriate value. Note that unrealistic values will cause iteration failure and error messages. Also supersonic flow is inhibited and will be reported as an error.

Note that there are special gas path components which have exceptions to these rules or include additional internal stations or locations for calculation static conditions:
- Inlet

The inlet has no upstream gas path component for calculating inlet station static conditions so a separate inlet static parameter edit box is added above the exit one.
- Exhaust nozzle

The exhaust nozzle inlet static conditions can be calculated by the upstream component. For the throat and exit stations, dedicated models are used to calculate static conditions including supersonic conditions in case of a con-di nozzle.
- Combustor

The combustor has a separate `Burner static conditions Duct cross area` for enabling calculation of averaged static conditions inside the combustor (between the inlet and exit stations) required for [Fundamental pressure loss] calculation and for afterburner mode efficiency map reading.

- [Mixer]

The mixer calculates separate mixing plane static conditions, required to calculate the equations relating the two mixing flows (usually using conservation of momentum and assuming a user specified relation between the two entry flow static conditions). Only the mixing plane exit corresponds to the component exit engine station.

8.4.1.1.1.2 Pressure loss models

The following standard GSP components include pressure loss models:

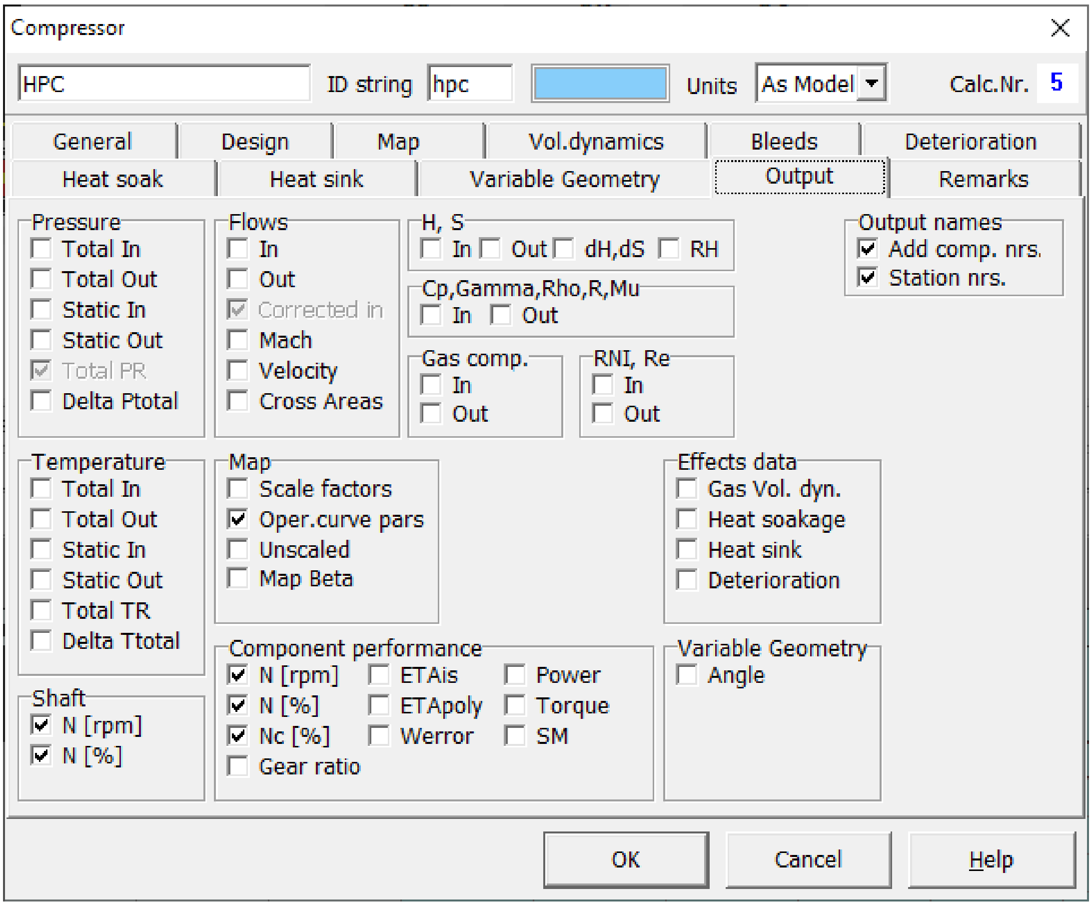
- *Inlet*
  In the inlet, aerodynamic pressure loss (due to friction) is represented by the ram recovery factor RR which is equal to `(1-dP/Pt)` with `dP/Pt` as relative pressure loss.
- *Duct*
  Duct aerodynamic pressure loss is represented by relative pressure loss functions or maps.
- *Combustor*
  In the combustor two types of pressure loss can be specified:
  - aerodynamic pressure loss represented by relative pressure loss functions or maps,
  - fundamental pressure loss determined from the increase in flow momentum occurring with temperature rise in the flow due to acceleration of the gasses.
- *Heat exchanger*
  Separate pressure losses can be specified in the two flow passages of the heat exchanger / recuperator component model, both aerodynamic and fundamental similar to the combustor component.

With relative pressure loss specified in the design point, several options exist to determine off-design pressure loss:

- *User specified (off-design)*
  Relative pressure loss remains equal to the design value unless the user specifies an off-design value.
- *User specified design only*
  Relative off-design pressure loss is calculated from the design value and the corrected entry mass flow maintaining a proportional relation between relative pressure loss and squared corrected entry mass flow (i.e. proportional to density*velocity2). Off-design pressure loss is not user specified directly!
- *Relative pressure loss map*
  Relative pressure loss is obtained from a map. The map type depends on the component (see the sections on the particular component). Off-design pressure loss is not user specified directly!

8.4.1.1.1.3 Common output parameters

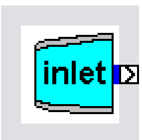
Component development in GSP uses window inheritance. This implies that components inherit the code from ancestor components. This also applies to all the elements on the component data windows of the component. An example is the `Output` parameter options tab sheet.



This tab sheet shows the main output options that are common for all gas path components. When the output options are checked, the output parameter will be written to the output table. The tab sheet contains options for pressures (`Pressure`), temperatures (`Temperature`), flows (`Flows`), enthalpy and entropy (`H, S`), additional gas properties (`Cp, Gamma, Rho, R, Mu`), gas composition (`Gas comp.`) (species must be added first trough the Output options window), Reynolds output (`RNI, Re`), Map data (`Map`), bleed ports (`Bleeds`) and other effects data (`Effects data`).

Specific output data will be added by the derivatives of this model component.

8.4.1.1.2 Inlet



The inlet component represents all types of inlets used for gas turbines. The inlet requires specification of the gas turbine design point inlet air mass flow and specification of pressure loss in the form of the ram recovery factor RR. RR represents the ratio of total pressure at inlet exit (usually fan or compressor entry) divided by free stream (ambient) total pressure. Four options are available for specification of the ram recovery factor:

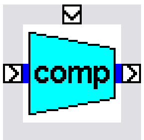
- `MIL-E-5008B standard`
  RR according to this standard assumes RR=1 up to flight Mach 1. Above Mach 1, RR is reduced using a function of Mach.
  Note that the MIL-E-5008B standard relation is scaled to design RR. This means RR values deviating from MIL-E-5008B standard may result if the design RR deviates from the MIL-E-5008B standard at the design Mach number. The Ambient conditions information box in the

Design tab sheet indicates design flight Mach number and other ambient condition information.
Warning: A design RR value higher than the MIL-E-5008B standard value (e.g. with design flight Mach number larger than 1) may result in (unrealistic) subsonic RR values higher than 1.

- `User specified RR`
  Both design and off-desing RR are user specified.
- `User specified RR design only`
  Design RR is user specified, off-design RR is corrected assuming linear relation between relative pressure loss (1-RR) and squared corrected inlet massflow.
- `RR map`
  Ram recovery determined by an inlet map. The map file consists of tables with RR as a function of flight mach number and corrected airflow at inlet exit. View the map graph using the `Map` button in the `Map` tab sheet.

In the Design tabsheet, design RR / PR can be specified. The `Set to MIL Std` button can be used to set the design RR to the MIL-E-5008B standard value (function of flight Mach number).

8.4.1.1.3 Compressor



The compressor component represents various types of compressors used in the main gas path except for the Fan. Usually the compressor is driven by a turbine and consequently shares a shaft number with it. Design point input includes rotor speed, pressure ratio and efficiency. Design inlet mass flow is obtained from the exit mass flow of the preceding component (such as an Inlet, Fan or other Compressor). Specify a shaft number or suffix corresponding to the engine specification, such as 2 for a typical gas generator compressor resulting in `N2` for the rotor speed parameter name.

Three simulation options for the compressor exist:

- `Free state rotor speed`
  The standard option, where the compressor rotor speed is a "free state variable" in the model dependent on the system performance. steady state rotor speed then is an equilibrium value where the compressor has just sufficient power to maintain rotor speed. During transient the compressor will accelerate or decelerate depending on the available shaft power.
- `User specified rotor speed`
  In this case the rotor is fixed to a user specified value. With this option power required is calculated for the specified speed. All other components (e.g. turbines) must have this option set also. GSP simulation output usually reports a power surplus or deficit for the shaft. This option is necessary to perform steady state simulation of single shaft turboprop or turboshaft engines which usually have a system where the shaft load is varied to maintain rotor speed. The operating line uses the same map format as the normal component maps, the only differences are that a surge line will not be required and that there is only a single value for Beta (by definition 1).
- `Externally controlled rotor speed`
  This option allows other components (custom components) to set rotor speed explicitly. The compressor and fan components and descendant component model classes need the 'Externally controlled' option set when the shaft speed is not a free state and determined by another component such as a turbine with the 'Power balance at rotor speed' option set.

For off-design performance several options are available. Using the radiogroup box one of the following options can be chosen:

- `Map`
  The standard option uses a compressor map of type component map. The map file consists of tables with corrected mass flow, efficiency and pressure ratio as a function of corrected normalised rotational speed and beta. The map operating point corresponding to the design operating point is specified using the map design rotational speed and beta values. Click the `Show Graph` button on the `Map` tab sheet to view the map graphically and note the yellow rectangle which can be used to move the map design point. In the map graph also the beta-lines can be show after activating the appropriate item in the `Options` menu.
- `Operating Line`
  This advanced option allows the engine modeller to use the compressor operating line. In case the maps of the compressors cannot be obtained from the manufacturer, the operating line of an existing engine can inserted (or a predicted operating line for a new engine). When using this option note that the mass flow error equation for the turbine needs to be deactivated to obtain a solvable equation system. In case the operating line is the actual running line of the engine, the error will remain very small (can be visualised through the option Werror on the turbine output tab sheet). Using a predicted operating line, the error increases the further the operating point is located from the real operating point.
- `No map (DP only)`
  This advanced option allows the engine modeller to calculate design point performance without having to specify an off-design component map. Off-design calculation will not be possible.

The map file consists of tables with corrected mass flow, efficiency and pressure ratio as a function of corrected normalised rotational speed and beta. The map operating point corresponding to the design operating point is specified using the map design rotational speed and beta values. Click the `Graph` button in the `Map` tab sheet to view the map graphically and note the yellow rectangle which can be used to move the map design point. In the map graph also the beta-lines can be show after activating the appropriate item in the `Options` menu.

An unlimited number of compressor bleed flows can be specified in the `Bleeds` tab sheet. Compressor bleed flows are secondary air flows extracted from the compressor air flow for various purposes: bleed flows can be used to model customer bleeds or turbine cooling bleeds used in the turbine component or air flows entering bypass ducts in the duct component. The bleed flow numbers used must be unique for the model.

The option `Reset map scale factors` sets the map scaling factors to 1 during off-design calculations. This enables the engine modeller to match existing turbo components using their actual/original maps.

When using a map, the surge margin (parameter name SM) is calculated during iteration (and not only at final output as an additional output value) to enable direct control of SM using limiters and schedulers.

Bleed control and enabling/disabling cells in (compressor) bleeds table depending on case or config. Bleeds can only be defined in configs, in cases, only bleed values and bleed control nrs. can be changed.

Changes in geometry caused by inlet guide vanes or variable stator vanes can be modeled using tab sheet `Variable Geometry`. Two options are available to model variable geometry, using a schedule on the tab sheet, or using an external geometry controller. For both options the `Effects per degree vane- or blade angle` group box needs to be populated with reference values for the modifiers for the map, where dAngle is deviation in degrees of reference vane/blade position. There are modifiers for mass flow, pressure ratio and efficiency. Note that when using the schedule the values are ascending.

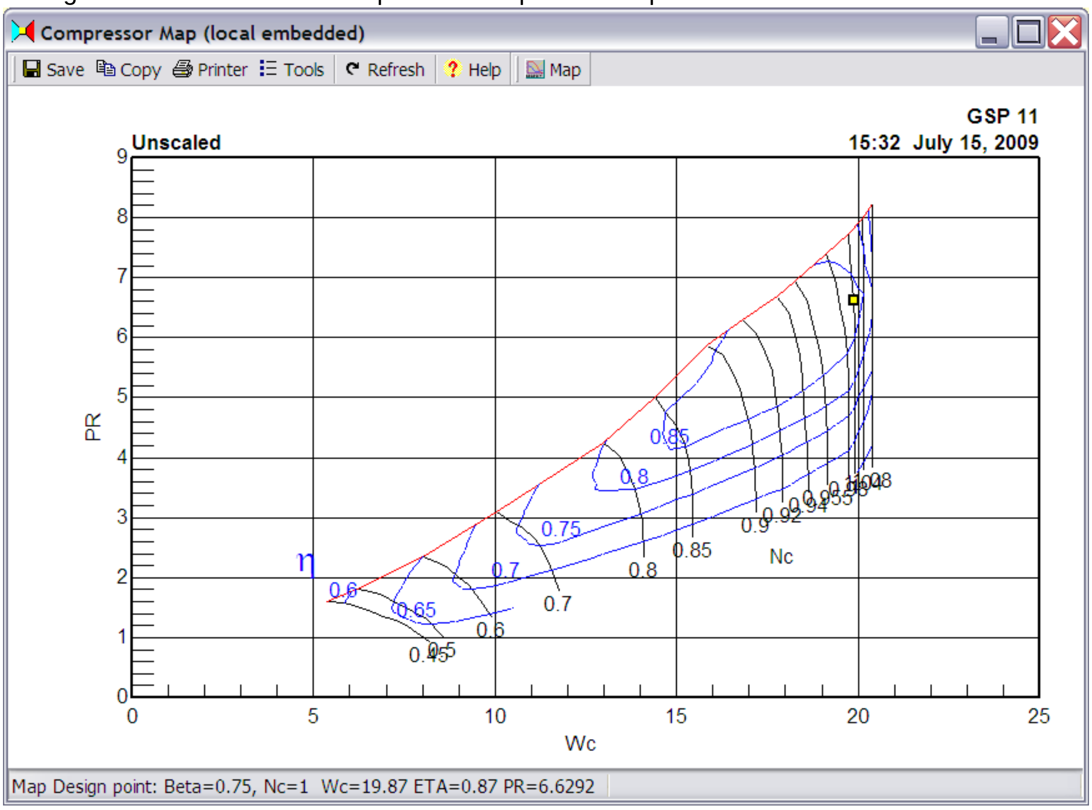
Modelers can model deterioration using the map modifying parameters on the `Deterioration` tab sheet. There are modifiers for mass flow, pressure ratio and efficiency. Note that to enable deterioration effects in your model, the `Apply deterioration effects` option must be active, which can only be used in off-design analysis.

Heat sink and heat soakage heat fluxes affect compressor work significantly. See Heat transfer on how this effect is modeled.

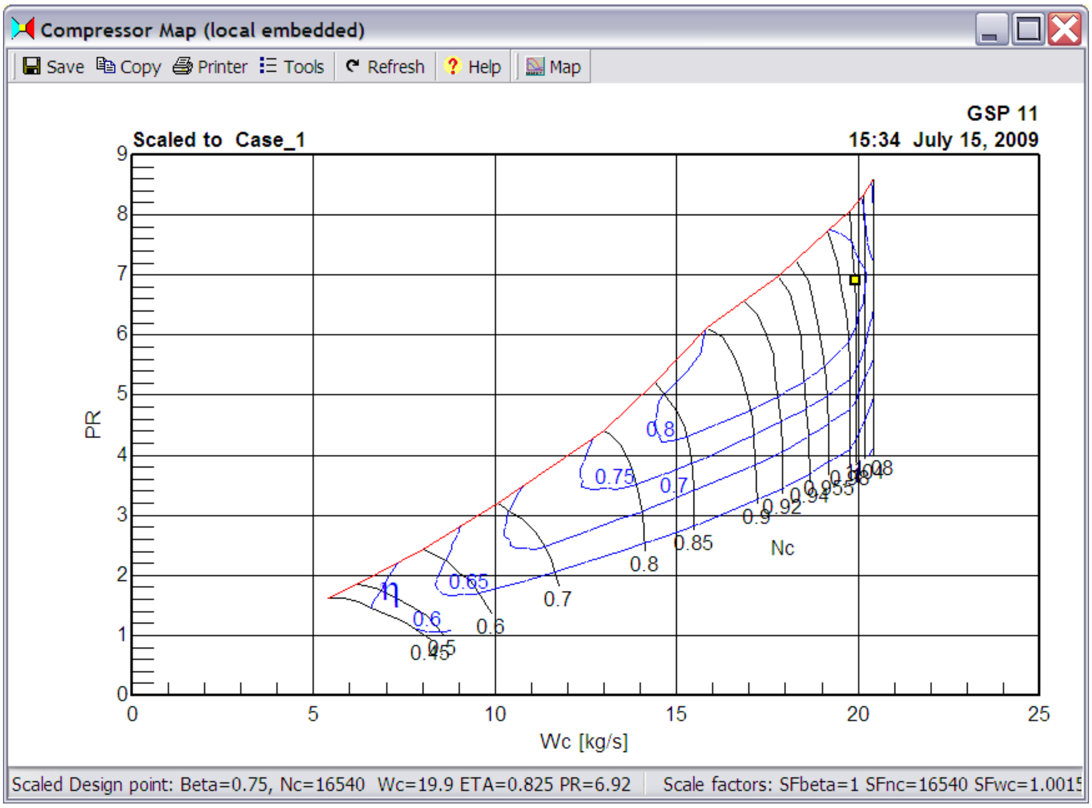
8.4.1.1.3.1 Compressor map

The compressor characteristics are represented in the compressor component map. The map displays the relation between 4 parameters corrected for entry conditions: corrected rotor speed `Nc`, pressure ratio `PR`, corrected mass flow `Wc` and isentropic efficiency `ETAc`. In GSP, the beta parameter composed of `Wc` and `PR` is used for better simulation stability.

The figure below shows an example of a compressor map



Note that the little yellow rectangle represents the map design point (DP) to which the engine design point is scaled. If the map graph is scaled using the `Map | Scale to model design point` menu option, then naturally the rectangle represents the actual engine DP. See example of a scaled compressor map below.



8.4.1.1.3.2 Operating Line

To assess the off-design behaviour of gas turbine components, component maps are used to define the off-design characteristics. When the components are grouped to form a gas turbine the range of possible operating conditions depends on the equilibrium of the gas turbine as a whole. The equilibrium points can be plotted in the component charateristics map to form the equilibrium operating (running) line.

To demonstrate the operating line usage in the compressor, a running line for a TJET model has been constructed using a decreasing fuel flow sweep. Figure 1 shows the compressor off-design charateristics for the TJET. The blue line denotes the actual (real) compressor running line. Imagine that we do not know the operating line and assume that the operating line is linear (red line).

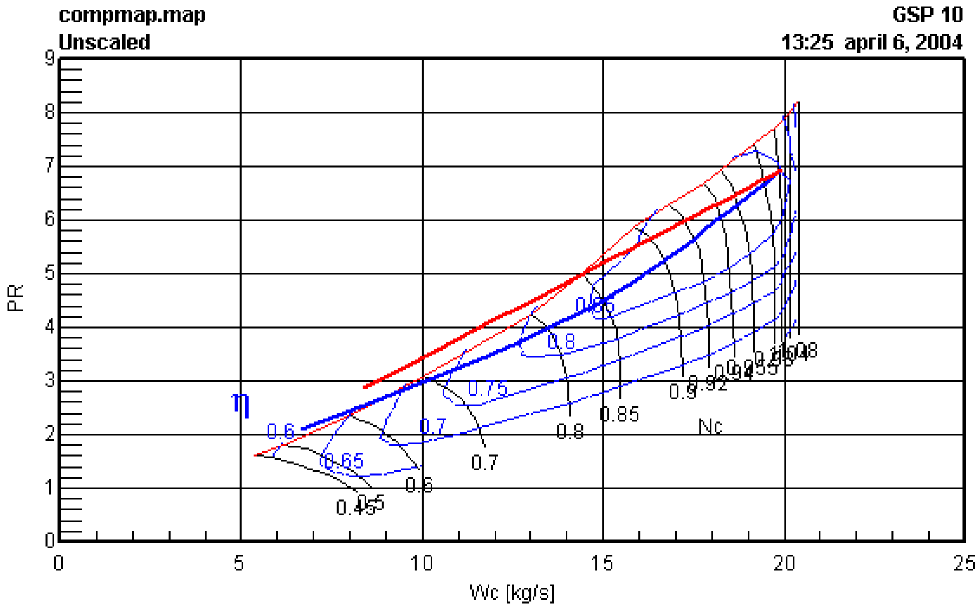


figure 1 Compressor Map; actual TJET compressor running line (blue line), assumed TJET compressor operating line (red line)

Since the option "No Massflow Error Equation" needs to be checked to deactivate the turbine massflow error equation (to obtain a solvable equation system) the error increases the further the asumed operating line deviates from the actual operating line. In case the operating line is the actual running line of the engine (or close to), the error will remain very small (can be visualised through the option Werror on the turbine output tab sheet). Figure 2 shows the massflow error for the actual and the assumed operating line.

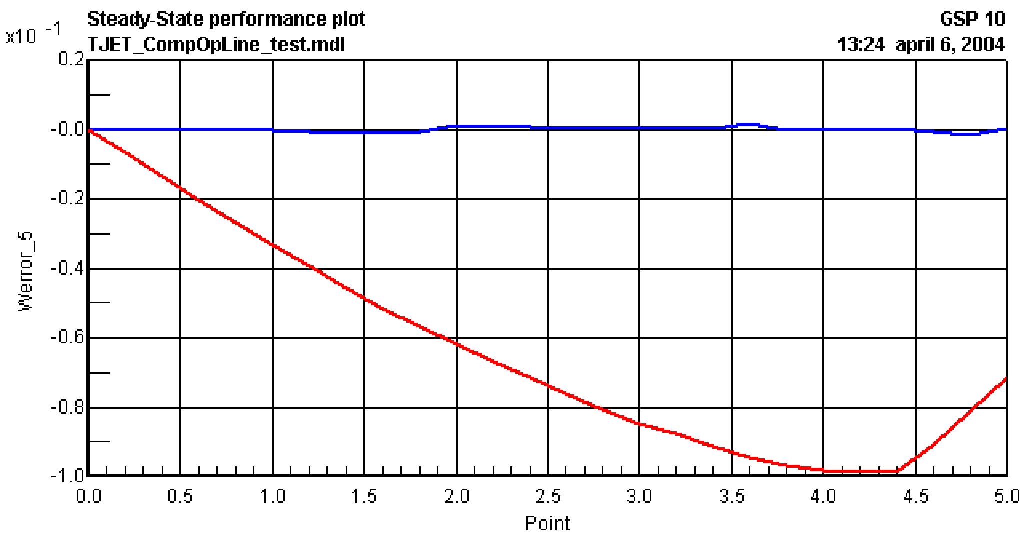


Figure 2 Turbine massflow error result

Figure 2 clearly shows that the further the assumed operating line is away from the actual operating line the massflow error of the turbine increases.

8.4.1.1.3.3 Surge margin

The surge margin (SM) is defined as the distance between the surge line and the operating point on a vertical line for a constant corrected mass flow value. Several definitions are in use to define the SM. From the compressor map it is clear that the distance to the surge line can be defined using multiple definitions. The most commonly used definition is defined by SAE, where the SM is defined as the distance from the operating point to the surge line at constant flow:

$$SM = 100 \cdot \frac{PR_{surge} - PR_{operating\quad line}}{PR_{operating\quad line}} = 100 \cdot \left(\frac{PR_{surge}}{PR_{operating\quad line}} - 1 \right)$$

This definition for SM is used by GSP.

An alternative definition, that is more oriented at the physical surge process is defined at constant speed (thus following the constant speed line to obtain the PR surge), is not used by GSP:

$$SM_{N=c} = 100 \cdot \frac{W_{operating\quad line}}{W_{surge}} \cdot \frac{PR_{surge}}{PR_{operating\quad line}}$$

Note that there are definitions around where the pressure ratio is corrected by subtracting 1 from the PR to account for pressure ratios being larger than 1 (see reference [1]) e.g.:

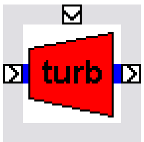
$$SM = 100 \cdot \frac{PR_{surge} - 1}{PR_{operating\quad line} - 1}$$

The pressure ratio used in the SM calculation is the PR wherein scaling effects for deterioration and variable geometry have been taken into account. This means that the SM for operating points that use these effects is not very accurate.

The required SM varies between different applications and is highly depending on engine configuration, accel and decel times, inlet distortion, compressor type (LP, HP, centrifugal, axial), etc. Typical values for the SM are in the order of 10 - 20% for fans,10 - 25% for LP compressors, and 15 to 30% for HP compressors, depending on the type of application. For a fan the inlet distortion alone contributes with 5% to the total fan SM.

Surge margin is scaled in scaled maps corresponding to the scaling of pressure ratio, i.e. proportional to PR-1.

8.4.1.1.4 Turbine



The turbine component is used to obtain mechanical power from hot gas in order to drive a rotating shaft and represents various types of turbines used in the main gas path. When the turbine drives a compressor it must share a shaft number with it. For a power turbine any shaft number can be used. If transient simulations are to be performed the rotor moment of inertia must be specified. The turbine moment of inertia applies to the entire spool (including any compressors, gearboxes and load components (e.g. generators, propellers etc.) rotating with it). Mechanical efficiency also applies to the entire spool and determines how much power is lost before it is extracted by a compressor or load outside the gas turbine.
Turbine design point input includes rotor speed and efficiency. Design inlet mass flow is obtained from the exit mass flow of the preceding component (such as a combustor or other turbine). Specify a shaft number or suffix corresponding to the engine specification, such as 2 for a typical gas generator compressor resulting in `N2` for the rotor speed parameter name. Design pressure ratio is not specified but calculated from the power required by compressors, fans and external turbine design loads in the design point calculation.

A series of turbine cooling flows can be defined to accurately model the thermodynamic effects of turbine cooling, including separate nozzle guide vane cooling (note that you can change the station number of the nozzle guide vane outlet on the `output` tab sheet).

For the determination of the rotor speed three options are available (also see compressor and turbo shaft models), user-linked to the options of the fan and compressor:

- `Free state rotor speed`
  The rotor speed is determined with use of the other components in the iteration process of the model. If the `free power turbine` checkbox is checked, no compressor is driven and the speed is determined by torque surplus and spool moment of inertia (causing acceleration or deceleration). Note that free power turbine speed only changes during transient simulation!
- `User specified rotor speed`
  The rotor speed is user specified and fixed. This option can be used for both Gas Generator (GG) and Free Power Turbine (PT) turbines. In case of a free PT, the `free power turbine` checkbox must be checked, see also free power turbines. Note that enforcing the rotational spool speed of the GG turbine (the model option of the compressor of the GG must also have the `User specified rotor speed` model option selected) may result in a power surplus or deficit for the shaft.
- `Speed determined by shaft (external control)`
  The rotor speed is user specified and the user can add an error variable (check box Add Power Balance equation, see also turbo shaft models) for the turbine power balance to calculate other components' (free state) parameters such as fuel flow or a control system variable (e.g. trimming signal) with the specified rotor speed as input. Any compressors or fans attached to this turbine must then have the `Externally controlled` option set.

GSP is able to calculate through a series of turbines (or stages; provided that stage maps can be obtained from the turbine manufacturer) to drive a single shaft. However, the user should provide the model the division of the overall power over the series of turbines (or stages) at design time. To accomplish this the user can choose from 6 options in the `Power delivered to shaft in design point` group box to divide the power per turbine:

- `All required`
  Used for the last turbine or a single turbine on one shaft; all power to shaft,
- `Part of req. pwr.`
  Specify the fraction of the total (all turbines) power to be given to the shaft for this turbine,
- `Power`
  Specify the amount of power from this turbine to be given to the shaft,
- `Torque`
  Specify the torque from this turbine to be given to the shaft,
- `PR`
  Specify the pressure ratio of this turbine determining the power to be given to the shaft,
- `TR`
  Specify the temperature ratio of this turbine determining the power to be given to the shaft.

Note that the in case there are multiple turbines placed on the same shaft the mass moment of inertia is taken from the last turbine on the shaft (with the highest component ID). Also note that the turbine moment of inertia applies to the entire spool including any compressors, gearboxes and load components (e.g. generators, propellers etc.) rotating with it (see above).

For both the compressor drive and free power drive option, (extra) power may be absorbed from the shaft using the `PTO` power and `Torque` specification fields. In this case it is also possible to use the Load Control component. Note that `PTO` power and `Torque` can be specified, which are summed.

The checkbox `Calculate max. Design load` of the `Design External load / PTO` box is very useful when modeling industrial gasturbines. When checked, the maximum amount of power that can be generated by the turbine will be calculated. The relative losses in the exhaust system can be given in the field `ass. exit to ambient rel. press.loss` to directly account for the exhaust pressure losses during design calculation.

For off-design performance several options are available. Using the option group box one of the following options can be chosen:

- `Map`
  The standard option uses a turbine map of type component map. The map file consists of tables with corrected mass flow, efficiency and pressure ratio as a function of corrected normalized rotational speed and beta. The map operating point corresponding to the design operating point is specified using the map design rotational speed and beta values. Click the `Show Graph` button on the `Map` tab sheet to view the map graphically and note the yellow rectangle which can be used to move the map design point. In the map graph also the beta-lines can be show after activating the appropriate item in the `Options` menu.
- `No map (DP only)`
  This advanced option allows the engine modeller to calculate design point performance without having to specify an off-design component map.

The option `Reset map scale factors` sets the map scaling factors to 1 during off-design calculations. This enables the engine modeler to match existing turbo components using their actual/original maps.

`Variable Geometry` and `Deterioration` are similar to the functionality described in the Compressor section.

Heat sink and heat soakage heat fluxes affect turbine work significantly. See Heat transfer on how this effect is modeled.

8.4.1.1.4.1 NGV cooling

A single nozzle guide vane (NGV) cooling flow can be defined in the `Cooling` tab sheet of the turbine. NGV cooling flow is assumed to represent the flow cooling the 1st stage stator vanes of the turbine (in case a multi-stage turbine is simulated by the GSP turbine component).

NGV cooling flow is entirely mixed with the turbine flow BEFORE turbine work is extracted during the expansion process. The NGV cooling flow therefore always fully contributes to the extracted turbine work, where for the other bleed flows (which are mixed just before turbine exit) the contribution to the delivered work is dependent on the `Press. Frac.` parameter.

The following data must be specified in the single available row:

- `Bleed flow nr`
  The compressor bleed flow number providing the cooling flow. This number must represent an existing compressor bleed flow, defined in a compressor component.
- `Frac. for cool.`
  The fraction of the compressor bleed flow used for this turbine cooling flow.
- `Frac. Eff. T.flow`
  The fraction of the cooling flow that contributes to the turbine effective flow. A larger or smaller part of the flow can be added to the flow obtained from the flow - pressure ratio relation in the map. A fraction of 1 means all cooling flow requires a proportional share of the cross flow area, at the cost of the turbine entry flow (e.g. the main flow exiting the combustor) so that for the same pressure ratio the entry flow is smaller. This would be the case if all cooling flow enters the turbine at the entry. If all cooling flow would enter at the exit, the fraction should be set to 0. Usually this fraction value is set between 0 and 1 and used to accurately fine-tune the model to known engine data.

8.4.1.1.4.2 Turbine cooling

Turbine cooling flows can be defined to accurately model the thermodynamic effects of turbine cooling. In the `Cooling` tab sheet an unlimited number of cooling flows can be entered, each row representing data of a separate cooling flow. The following data must be specified in the subsequent columns:

- `Nr`
  The cooling flow number to identify the cooling flow with a unique number in the model. Note that these numbers do not correspond with compressor bleed flow numbers.
- `Bleed flow nr`
  The compressor bleed flow number providing the cooling flow. This number must represent an existing compressor bleed flow, defined in a compressor component.
- `Frac. for cool.`
  The fraction of the compressor bleed flow used for this turbine cooling flow.
- `Frac. Eff. T.flow`
  The fraction of the cooling flow that contributes to the turbine effective flow. A larger or smaller part of the flow can be added to the flow obtained from the flow - pressure ratio relation in the map. A fraction of 1 means all cooling flow requires a proportional share of the cross flow area, at the cost of the turbine entry flow (e.g. the main flow exiting the combustor) so that for the same pressure ratio the entry flow is smaller. This would be the case if all cooling flow enters the turbine at the entry. If all cooling flow would enter at the exit, the fraction should be set to 0. Usually this fraction value is set between 0 and 1 and used to accurately fine-tune the model to known engine data.
- `Press. Frac.`

This fraction represents the extent to which the cooling flow contributes to the expansion process providing mechanical power in the turbine. At 0, no expansion power is provided. At value 1, the cooling flow entirely expands with a pressure ratio equal to the turbine pressure ratio, starting with the (compressor) bleed flow temperature. Usually this fraction value is set between 0 and 1 and used to accurately fine-tune the model to known engine data.

- `Exit radius [m]`
  This is a radius for calculation of the effect of the increase in kinetic energy of the cooling flow due to the increase in velocity of the flow passages with increasing distance (radius) from the rotating shaft centerline. The kinetic energy is added to the cooling flow enthalpy and subtracted from the turbine shaft power. The radius is specified in meters and usually is set equal to the average distance of the blades to the shaft centerline. This value can be used to accurately fine-tune the model to known engine data.

Turbine cooling flow output parameters are identified by the turbine **local** cooling flow number (e.g. `Wcl3_6` is mass flow rate of cooling flow nr. 3 in turbine component nr. 6).

8.4.1.1.4.3 Free power turbine

Click the `free power turbine` option for a free power turbine not driving a compressor. If a free power turbine also has the `Free state rotor speed` option set, the model should be used for transient calculation only to simulate free power turbine accels and decels with a power turbine speed governor. Use user specified rotor speed for steady state calculations with free power turbine to prevent GSP from searching a steady state power turbine speed / load combination which often does not exist (power turbine either stops or overspeeds).

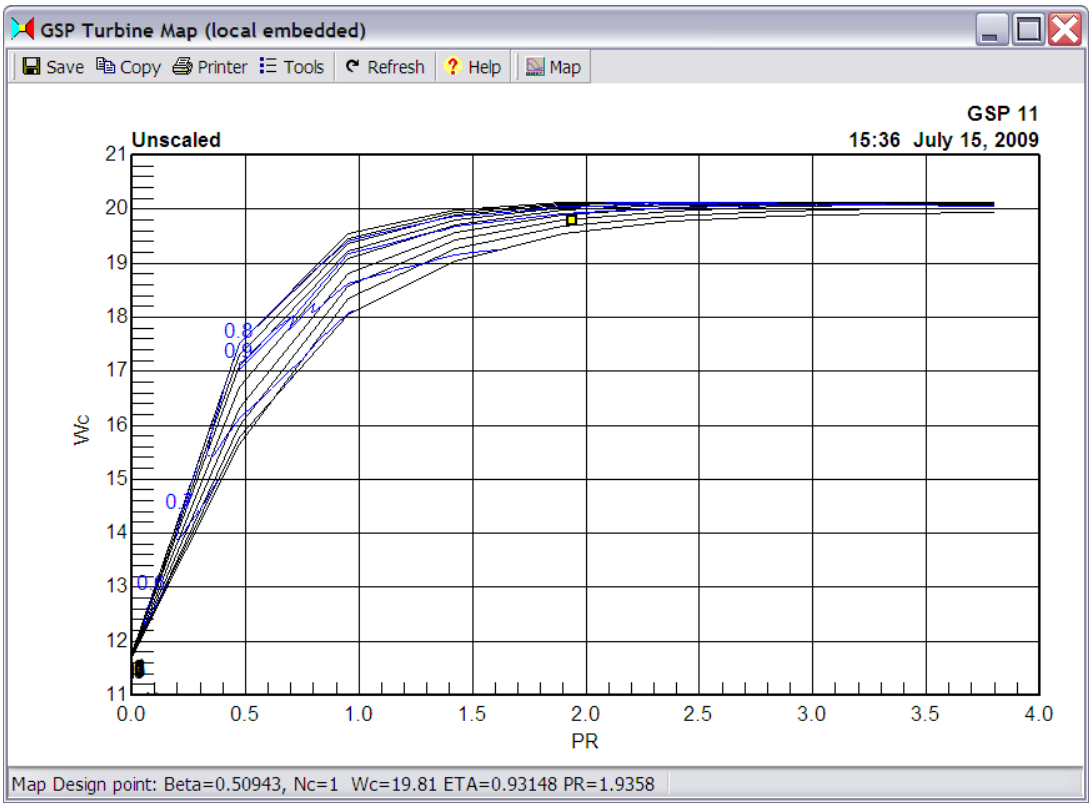
For off-design performance a component map is used. The map file consists of tables with corrected mass flow, efficiency and pressure ratio as a function of corrected normalized rotational speed and beta. The map operating point corresponding to the design operating point is specified using the map design rotational speed and beta values. Click the `Graph` button in the `Map` tab sheet to view the map graphically and note the yellow rectangle which can be used to change the map design point. In the map graph also the beta-lines can be show after activating the appropriate item in the `Options` menu.

For both the compressor drive and free power drive option, (extra) power may be absorbed from the shaft using the `PTO` power and `Torque` specification fields. In this case it is also possible to use the Power Turbine Load Control component.

8.4.1.1.4.4 Turbine map

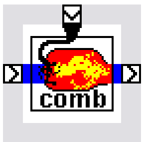
Turbine performance characteristics are represented in the turbine component map. The map displays the relation between 4 parameters corrected for entry conditions: corrected rotor speed `Nc`, pressure ratio `PR`, corrected mass flow `Wc` and isentropic efficiency `ETAc`. In GSP, the beta parameter represents turbine `PR`.

The figure below shows an example of a turbine map



Note that the little yellow rectangle represents the map design point (DP) to which the engine design point is scaled. If the map graph is scaled using the `Map | Scale to model design point` menu option, then naturally the rectangle represents the actual engine DP.

8.4.1.1.5 Combustor



The combustor component is used to simulate both primary (gas generator) combustors and afterburners. Data may be specified for:

The combustion process

- Fuel type and/or composition
  There is a distinction in design and off-design fuel type. This is to analyze OD effects of alternative fuels for example.

- Combustor design fuel input (Design tab sheet), specified by either
  - `Fuel flow Wf`
  - `Exit temperature`
  - `Fuel-Air Ratio`
  - `Stator Outlet Temp SOT`

  For the latter 3 options, corresponding fuel flow is calculated automatically using GSP's chemical combustor model maintaining full conservation of energy. The input field corresponding to the selected input is active for input, the other 3 disabled. The `Update input to DP` button resets the inactive input fields to the last calculated Design point value if existing. This is convenient to directly see corresponding values and after switching input type to have the correct value for the new parameter to be used.

For the last (SOT) option, a DP equation has to be added. SOT is evaluated in an error equation (with Wf as state variable) for the iteration towards user specified SOT (iteration necessary because SOT may be affected by downstream hpt cooling flows).

- Combustion efficiency, using one of a number of different models for off-design efficiency
  - `User specified combustion efficiency`
    User defined fixed off-design combustion efficiency
  - `Use combustion efficiency map`
    Efficiency as function of combustor temperature rise and pressure ratio Delta
  - `Use afterburner combustion efficiency maps`
    Define 3 maps to determine afterburner combustion efficiency based on FAR, and corrections for Mach number and relative pressure drop:
    1. `Comb. eff. vs. FAR map text file`, to calculate the efficiency from reheat FAR
    2. `Flow Mach nr. correction map text file`, to calculate the efficiency correction for Mach value
    3. `Pressure correction map text file`, to calculate the correction factor for relative pressure drop

Combustor pressure loss

Different models for off-design pressure loss are modelled:

- `Specified design rel. pressure loss only`

User specified design pressure loss only, with off-design calculated relative pressure loss (=PR) by scaling to corrected mass flow (squared)

- `User specified off-design press. loss`

Use fixed user specified off-design pressure loss

- `Pressure loss map`

Use pressure loss map, dP = f(Wc), where Wc is based on

- `Corrected entry mass flow`
- `Fuel mass flow`

Pressure loss as result of the addition of heat and resulting increase in velocity:

- `Calculate Fundamental Pressure Loss`
  The fundamental pressure loss is determined with the conservation of momentum and is usually used for afterburner mode only, when the effect becomes significant due to the very high temperature increase.

Emission formation

Optionally combustor exit emission values (NOx, CO, UHC indices and Smoke number) can be calculated using either one of three emission formation models:

- `None`
- `Interpolation in ICAO table (NLR correction method)`
- `Semi-empirical ratio- or direct prediction method`

Note that an additional option can be found in the component Multi Reactor Combustor found in the Special Gas Path Component Library inheriting from Combustor.

- `Multi-reactor combustion model`

Static conditions inside the burner

The combustor has a separate `Burner static conditions Duct cross area` for enabling calculation of averaged static conditions inside the combustor (i.e. between the inlet and exit stations). These static conditions are required if Fundamental pressure loss calculation is required or when the combustor is running in afterburner mode and the static pressure input for the afterburner efficiency map is required.

Fuel pump/compressor

The power required for compressing the fuel for injection into the combustor can be calculated using the Fuel pump.

Water injection

To lower combustor temperature for e.g. lowering NOx emisions or to increase specific power output, water or steam can be injected for both design and off-design calculation.

Afterburner specific
Design point calculation frequently does not include afterburning, hence an option on the `Design` tab sheet has been added to set the design afterburner fuel flow to zero (Checkbox "`Zero Wf in design Calc. (afterburner)`").

`Output` tab sheet
The combustor component inherits most of the inlet and exit parameters of the Common output parameters, and adds combustor specific output parameters (`Combustion` and `Emissions`)

`ERchem` is the chemical equivalence ratio. This output parameter contains the calculated chemical equivalence ratio. This is simply the quotient of the total oxygen needed for a stoichiometric mixture and the total oxygen that is present in the mixture. For the difference between the chemical equivalence ratio and the more known equivalence ratio defined by the quotient of actual fuel-air-ratio and stoichiometric fuel-air-ratio, readers are referred to NASA RP1311, Users Manual. However, a few remarks are made here: if all the positive valence atoms (C,H,..) are present in the fuel and all the negative valence atoms (O,..) in the oxidizer, the two equivalence ratios are equal. If not, they are still equal when they are one (stoichiometric mixture), and they are both smaller than one for lean mixtures and higher than one for rich mixtures. The chemical equivalence ratio can be determined for a mixture, without prior knowledge of the fuel composition, for the other equivalence ratio, the fuel composition must be known.

`LHV` is the Lower Heating Value. This is the heat of combustion where it is assumed that the water component of the combustion process is in vapor state at the end of the combustion.

`Unburnt` is the amount of fuel flow that is not burnt in the combustor.

8.4.1.1.5.1 Combustion process

The combustion heat release is calculated from the fuel flow and fuel properties. Fuel properties can be specified in terms of a fuel type and H/C ratios and lower heating values. If desired, the fuel composition can be specified instead, composed of a number of predefined species. This option is to allow simulations of systems using alternative fuels such as gasified bio-mass.
Design fuel can be specified separately (from the off-design fuel) to allow maintaining a reference (design) fuel for calculating fuel effects.
Instead of fuel flow, also combustor exit total temperature of combustor exit fuel-air ratio can be user specified to determine the combustor operating point. In that case, fuel flow is calculated. After selecting the appropriate option in the `Design` tab sheet, GSP works with either fuel flow, exit temperature (or stator outlet temperature, SOT) or fuel-air ratio, both design and off-design. Specifying exit temperature for example is very convenient if calculations with constant (i.e. maximum) turbine entry temperature levels are needed.
Off-design fuel flow (or exit temperature or fuel-air ratio) is specified with control components such as the manual fuel control.

For the design point, specify the design combustion efficiency (ratio of effective heat release and 100% combustion theoretical heat release). During off-design the combustion efficiency is determined as user specified or from a map. For a primary combustor, use the map option with combustion efficiency as a function of temperature rise dT and pressure level in terms of d. For an afterburner, use the afterburner efficiency maps with three 2-dim maps (effects of relative (to design values) fuel air ratio, nominal flow Mach number and pressure).

The combustion process is calculated using gas and fuel composition data and the equations for chemical equilibrium, meaning that dissociation effects (CO2|CO and H2O|OH) are accounted for.

Note that the calculation of delta enthalpy, dH, over the combustor has not much meaning as the composition changes (each individual specie in the gas model is having another temperature at which the H equals zero). Even if all H zero temperature would be equal, dH still has not much meaning in a component like a combustor with detailed/chemical gas models, since both mass and composition changes. Conservation of energy is maintained including a variety of chemical reaction enthalpies. Of course, for an idea of dH that would be used for hand calculation, one could define a calculated column like cp3*(TT4-TT3).

8.4.1.1.5.2 Combustor pressure loss

Design point relative pressure loss is specified in the `Design` tab sheet. Off-design point pressure loss is determined depending on the settings in the `Pressure Loss` tab sheet. The first option `Specified design rel. pressure loss` only calculates off design relative pressure loss proportional to squared corrected entry mass flow (see pressure losses). The second option `User specified off-design press. loss` allows direct user specification of off-design pressure loss. The third option uses a pressure loss map with relative pressure loss either as a function of corrected entry mass flow or fuel flow.

Click the `Calculate Fundamental Pressure Loss` option to calculate additional fundamental pressure loss, especially significant in afterburners.

8.4.1.1.5.3 Emission formation

The `Emissions` tab sheet of the combustor component contains the emission models determining NOx, CO, UHC and soot during combustion. When using emission models (instead of `None`), three options are available:

- `Interpolation in ICAO table` (or NLR emission model)
  Data available from the ICAO emission databank can be entered in a table for interpolation to obtain a rough means to interpolate between combustor operating conditions for NOx, CO, UHC emission index data and Smoke number data. The ICAO table consists of four data sets of emission indices for take-off, climb out, approach and idle.

- `Semi-empirical ratio- or direct prediction method`
  For this method emission indices in the design point are used to predict the emission indices in off-design conditions ('P3T3 models').
  Note: a prediction method for UHC is not yet available, therefore the EIuhc will always be equal to 0 for this option

- `Multi-reactor combustion model` [4] (available is emission component library only)
  A combustion chamber model is built by dividing the combustion chamber liner volume into an array of reactors. In each reactor 4 flows can enter: the flow from the previous reactor, an oxidant flow coming from outside the liner, the fuel flow, and a flow of water/steam. These 4 flows are assumed to mix instantaneously and reach equilibrium at the reactor exit.
  In general, two types of emission formation are discerned: instantaneous formation in a flame and gradual formation throughout the combustion chamber. The instantaneously formed emissions are added to the total amount of emissions present so far. The gradual formation determines emission formation rate equations. The (equilibrium) temperature,

composition and the actual emission concentration at each reactor exit are used to calculate the emission formation rates, which are numerically integrated.
Four mechanisms of NOx formation are modelled: prompt NOx, fuel NOx, thermal NOx and NOx formation by the N2O mechanism. The amounts of prompt NOx and fuel NOx are calculated using empirical equations and supposed to be formed instantaneously in flames. Thermal NOx and NOx formed by the N2O mechanism are assumed to be formed throughout the combustion chamber.
For CO emmissions it is assumed that the fuel reacts instantaneously to CO (and H2O), and is subsequently (gradually) oxidised to CO2 by one chemical reaction.
For UHC emmissions, the fuel is converted to an amount of jet fuel or methane, depending on the fuel used. Both the jet fuel and methane are (partially) oxidised in the subsequent part of the combustion chamber.
For smoke (soot), the assumption is made that the soot particles are spherical. An empirical equation is used to predict formation, while kinetic-type expressions are used to calculate the smoke oxidation.

The last two emission models calculate emission indices (EI) relative to a user defined design point reference EI value, making GSP particularly valuable for emission sensitivity analysis.

Note: Using the combustor component for emission calculations requires thorough knowledge on the underlying theory and equations of the options applied.

8.4.1.1.5.4 NLR emission model

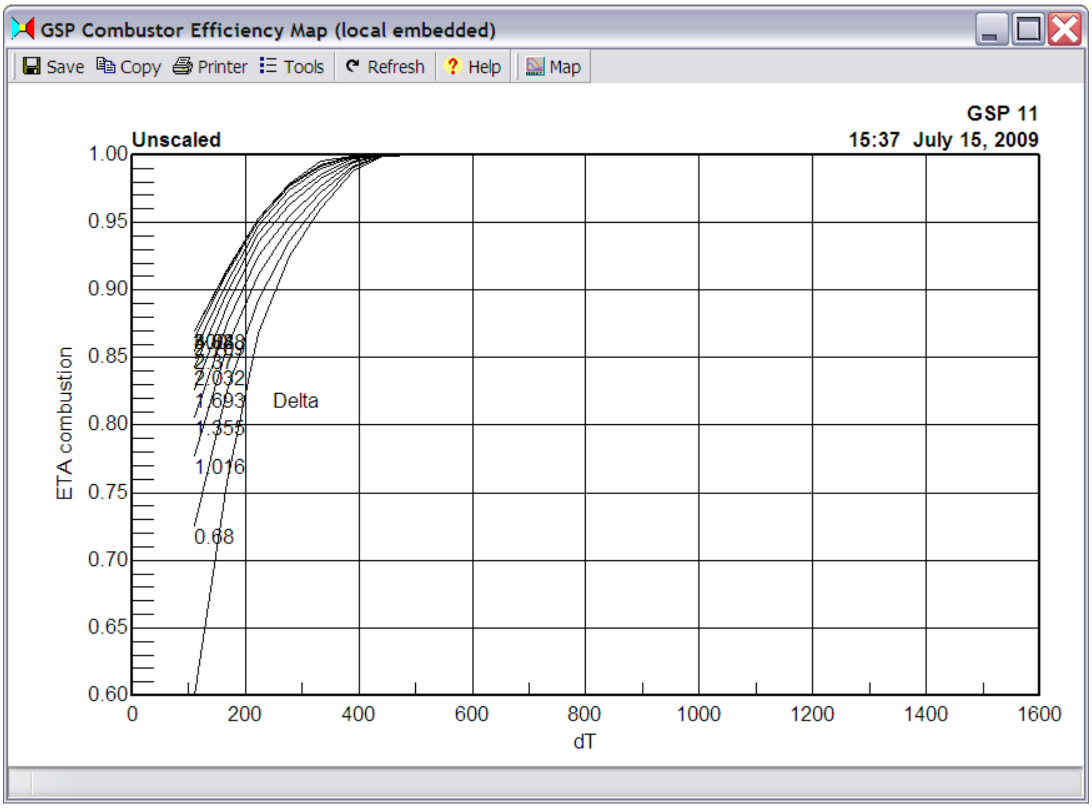
The NLR emission model is based on measured emission index values for new engines as published by ICAO for a large number of aero-engines.

The "NLR method" comprises of interpolation in tables derived from the ICAO thrust vs. emission index tables. For the specific engine, the combustor pressure and temperature are determined for each of the 4 ICAO defined thrust settings (this can be done using GSP). This data is entered in the table in the NLR/ICAO tab sheet in the combustor edit window. The emission index is then calculated using logarithmic interpolation with combustor temperature and a simple empirical correction for deviating combustor pressures.

Since the ICAO emission data are specified for standard conditions, application of this emission model is recommended for use only at standard operating conditions (standard ambient temperature and pressure and a standard undeteriorated engine).
For analysing effects of deviating operating conditions, it is best to use the ratio emission models.

8.4.1.1.5.5 Combustor map

Combustor performance characteristics can be represented in the combustor component maps. Combustor efficiency is obtained from a map if the appropriate option is selected in the combustor data entry window (per default, a constant user specified combustor efficiency is used). For primary combustors, combustor efficiency is represented in a special graph as shown below. For afterburner combustors (also an option), 3 alternate maps are used (see combustor component).



Combustor Combustor efficiency (ETA combustion) is a function of combustor temperature rise dT and Delta (Pt_in/Pstd). Only ETA combustion is scaled to a DP specified value.
In addition, pressure loss maps can be used to represent pressure loss as a function of squared corrected entry mass flow (Wcin^2).

8.4.1.1.5.6 Fuel pump

The power required for compressing the fuel for injection into the combustor can be calculated and taken from a user specified shaft. The fuel pump power effect can be separately specified and accounted for in DP and OD simulation mode. Pump/compressor power calculation depends on fuel type, either:

- `Incompressible (liquid) fuel flow`
  (power is pressure delta * volume flow, density needs to be specified)
- `Compressible with user specified ETAis`
  (power is calculated using the appropriate equation for compression power at a user specified isentropic efficiency)

The fuel pump/compressor entry pressure is user specified. The fuel pump/compressor exit pressure is static combustor inlet pressure plus an additional injection overpressure. If static combustor inlet pressure is not available (no entry area (i.e. upstream component exit area) specified) total is taken instead.
The fuel pump inlet temperature is taken from the Fuel (for DP Design Fuel) tab sheets user specified data.

The fuel pump calculation can be activated/deactivated (using the `Account for fuel pump/compressor power` check box) and works for both DP and OC calculations. Note that in order to analyse fuel compressor power effects as an OD effect, the Reinitialize model on case type change option in the `General | Initialization` options needs to be disabled: then run a DP first without Fuel pump active, then follow with OD with Fuel pump active.

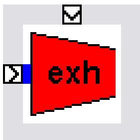
Calculation of absolute enthalpy (H) of fuel before (Hfueltank) and after fuel (Hfuelin) compression and output for use in system energy balance calculations:

a) When using Hfueltank for the energy balance, the fuel compression system is <u>included</u> in energy balance system boundary. Fuel compressor power and fuel temperature effects calculation options (`Add fuel temperature rise due to compression` and `Account for fuel pump/compressor power`) must be either both active or both <u>inactive</u> for the energy balance to remain consistent.

b) When using Hfuelin, the fuel compression sysem is <u>excluded</u> from energy balance system boundary. The fuel compressor power calculation option must be <u>deactivated</u>.

Note that the energy balance expression can easily be calculated as an output calculated expression. For a simple turboshaft for example as a % of total input heat rate:
100*(W1*H1+Wf*Hfuelin-W9.H9-PWshaft)/(Hv_b*Wf_b))
When the energy balance is closed properly, the result of this expression should be close to 0 (typically less than 0.1 %). For gaseous fuels, compression power and fuel enthalpy rise is calculated using the user specified isentropic fuel compression efficiency. For liquid fuels, fuel compression/punp power is calculated as volume flow rate * dp and the resulting work per kg fuel added as a 'virtual total temperature' of the liquid fuel to account for the energy released as kinetic energy in the fuel jets when exiting the injector and entering the combustor.

8.4.1.1.6 Exhaust nozzle



The exhaust nozzle component can be used for simulation of convergent nozzles, convergent/divergent nozzles with either fixed or variable geometry. Exhaust component models usually end the gas path of a GSP gas turbine model and simulate the expansion process of the gas to ambient or other user specified conditions. In the GSP equation system (unless the `Disable Massflow Error Equation` option is active), an exhaust component adds an error equation to enforce matching of the exhaust mass flow (calculated from exhaust gas conditions, flow cross-area and outside/ambient pressure) with the mass flow entering from the upstream component.
The variable geometry option allows control of either throat area only or both throat and exit area (in case of a convergent/divergent nozzle), using the nozzle control component. The exhaust nozzle component can be used for both thrust generating (jet-engine) nozzles including turboprop engine exhausts and turboshaft exhaust systems in which case thrust is minimal and not relevant. Exhaust stack pressure loss can be modeled by inserting a duct component.
Specify a velocity coefficient to represent losses in terms of a correction of the exit flow velocity resulting from isentropic expansion from entry to ambient pressure.

Note that for the design point, effective throat cross-flow area is not user specified, but calculated from the exhaust entry conditions, i.e. the area required for expansion to ambient conditions of the entry mass flow. Only by adapting component design data upstream of the exhaust nozzle component, design point nozzle area can be indirectly adapted. Decreasing design efficiency of an upstream turbine component for example will increase the area due to the decrease in turbine exit pressure resulting in lower density, requiring a large exhaust cross-flow area.
For a variable nozzle, off-design throat area can be directly specified using the nozzle control component, which effectively controls relative nozzle area (i.e. nozzle area/design nozzle area). Design point geometric

Thrust calculation

Ideal gross thrust is calculated using the customary equation for impulse reaction force of the flow plus the the static pressure difference from throat to ambient pressure times the throat area (only when choked):

FG_ideal = W*C_exit_ideal + A_exit*(Ps_exit-Po),

with C_exit_ideal, A_exit, and Ps_exit the ideal velocity, area and static pressure at the exhaust nozzle exit (either the convergent nozzle or con-di nozzle exit) and Po the ambient pressure.

Nozzle efficiency and losses

Two different ways are available to represent losses (efficiency) of the expansion process in the nozzle:

- *CV velocity coefficient*

  The CV factor represents the extent to which C_exit_ideal is reached after expansion. The difference between ideal and real exit velocity is due to viscous friction losses for example in the boundary layers.
  For the convergent nozzle, a CV value smaller than 1 results in a lower than ideal exit (i.e. throat) velocity and total pressure and higher exit static enthalpy Hs and temperature Ts (since the 0.5*C^2 term becomes smaller and total enthalpy remains constant). As a result, during design point calculations also the effective nozzle area A (usually A8) becomes little larger (do not confuse this effect with the CD discharge coefficient which represents the ratio between effective and geometric area). During off-design calculations, a smaller CV will decrease the flow rate. CV does not affect choking, only resulting throat exit velocity and static temperature are affected, exit static pressure is not affected by CV.
  For the con-di nozzle configuration, CV applies to the exit (usually station) of the divergent part (not the throat). In this case, area A9 is not affected, exit velocity gets lower and Hs, Ts higher with decreasing CV. Ps remains unchanged.
  Although the effect of CV as implemented in GSP cannot be simply seen as a factor reducing C_exit, the following equation will approximate the CV effect fairly well in many cases:

  FG = CX*(W*CV*C_exit_ideal + A_exit*(Ps_exit-Po))

- *CX thrust coefficient*

  CX is a much simpler method than CV to represent losses and is simply a factor multiplied with FG_ideal to yield actual thrust:

  FG = CX*FG_ideal

Although it is not recommended (usually either CV or CX is used), CV and CX can be applied in a combined fashion resulting in a relation approximated by:

FG = CX*(W*CV*C_exit_ideal + A_exit*(Ps_exit-Po))

CV and CX can be specified for Design Point and Off-design calculations separately. Note that design relative nozzle throat area (Aeff/Aeffdes) is defined 1.0. Relative nozzle area can be adapted in OD by the Manual Variable Exhaust Nozzle Control or derivatives of this control component.

Exhaust nozzle options:

`General`

`Model options`

Either choose a `Fixed area nozzle` or a `Variable area nozzle` to respectively model a fixed nozzle or a controlled exhaust nozzle. In the latter case a link appears on the component icon to connect an exhaust nozzle controller.

`Velocity coefficient CV`
Off-design CV (see above).

`Thrust coefficient CX`
Off-design CX (see above).

`Design`

`Convergent-Divergent nozzle`
Option to choose whether a convergent nozzle or a convergent-divergent nozzle is used.

`Ideal con-di nozzle complete expansion`
For con-di nozzle only, with this option on, complete ideal expansion from throat to ambient pressure is assumed (no shocks, no con-di nozzle calculations).

`Velocity coefficient CV`
Design CV (see above).

`Thrust coefficient CX`
Design CX (see above).

Note that normally either CX or CV are used since they more or less represent the same losses. Although it is possible, it is advised to not specify values deviating from 1 for both fields simultaneously.

`Specify area as`
- Specify effective areas and CD's (and have design geometric area/area ratio calculated). Also with a variable nozzle area nozzle, the nozzle control component controls *effective* throat area and (if option active) the con-di ratio as applied on *effective* areas. *Geometric* areas are calculated by dividing *effective* areas by the CD values.
- or specify geometric areas (and have design CD's calculated). Also with a variable nozzle area nozzle, the nozzle control component controls *geometric* throat area and (if option active) the con-di ratio as applied on *geometric* areas. *Effective* areas are calculated by multiplying *geometric* areas with the CD values.

`Throat`
- `CD`
  Discharge coefficient used to calculate the design geometric area of the throat
- `Ageom`
  Geometric area used to calculate throat throat CD in the design point used for subsequent DP and OD calculations.
- `Update to DP` (button)
  Update inactive input field values to last calculated Design Point.

`Exit`
- `CD`
  Discharge coefficient of the exit plane
- `Condi area ratio`
  Con-di area ratio of effective or geometric areas, depending on above described option.

`Disable Massflow Error Equation`
This option can be used to disable the mass flow error equation of the exhaust where Werror = f(Win, Wout). For advanced users only: when a state variable is removed from the system (e.g. map beta in case "no map" option is chosen in a compressor) somewhere an error equation has to be removed as well. The nozzle error equation is a sensible candidate for this.

Depending on the option settings, input fields are disabled where appropriate.

Output tab sheet

The `Output` tab sheet, for specification of the simulation output parameters, further contains 2 elements to specify the station numbers (default '8' for throat and '9' for exit). To have only output on the throat station (for a convergent nozzle), set the nozzle exit station number equal to throat (e.g. both '8'). Set to separate numbers for a con-di nozzle (e.g. '8' and '9').

Using the exhaust nozzle component

Take the following steps to configure a *Exhaust Nozzle* component in a GSP jet engine model:

1. Drag an *Exhaust Nozzle* from the GSP Gas Path Component Library to the model window.
2. Connect the exhaust to the last turbine, afterburner combustor, duct or heat exchanger.
3. Configure the exhaust as convergent or convergent-divergent and fixed or variable area nozzle.
4. Fill out the General - and Design tab sheets as described above.
5. Connect a nozzle control component if a variable nozzle is selected.

8.4.1.1.6.1 Convergent exhaust nozzle

In the convergent nozzle component type, isentropic expansion is calculated to ambient pressure, or up to sonic speed when the nozzle is choked. A velocity coefficient CV can be specified to represent losses via a reduction of the average jet velocity that is used to calculate the flow rate and thrust.
The throat area is not specified, but calculated in the Design point calculation, from the flow and pressure ratio resulting from up-stream calculated design point data.
For a convergent nozzle, the exit station (usually 9 for a hot exhaust) is defined to coincide with the throat station (usually 8 for a hot exhaust). Therefore all gas properties of station 8 and 9 will be equal.
A throat discharge coefficient CD (usually 'CD8') represents the relation between geometric and effective throat flow cross areas:

A_geom = CD * A_eff

The effective area is lower than geometric area due to boundary layer effects on the flow (reduced velocity due to friction).

In the design point A_eff is always directly determined by the upstream gas path calculation and A_geom derived using a user specified CD or directly user specified. Off-design either A_eff or A_geom can be user controlled depending on selected Exhaust nozzle options.

8.4.1.1.6.2 Con/Di exhaust nozzle

In the convergent-divergent nozzle component type isentropic expansion is calculated to ambient pressure if the nozzle is not choked. Expansion to the nozzle throat in the convergent part is similar to the convergent nozzle calculation. The expansion beyond the throat becomes supersonic if the nozzle throat is choked. Determined is wether the flow is under- or over-expanded. For over-expanded flow, the location of the normal shock is determined (in- or outside the divergent part of the nozzle). If the flow is over-expanded with the shock inside the nozzle, shock losses have to be accounted for and expansion to ambient pressure in the nozzle exit is calculated.

A nozzle exit discharge coefficient CD (usually 'CD9') represents the relation between geometric and effective exit (end of divergent part) flow cross areas:

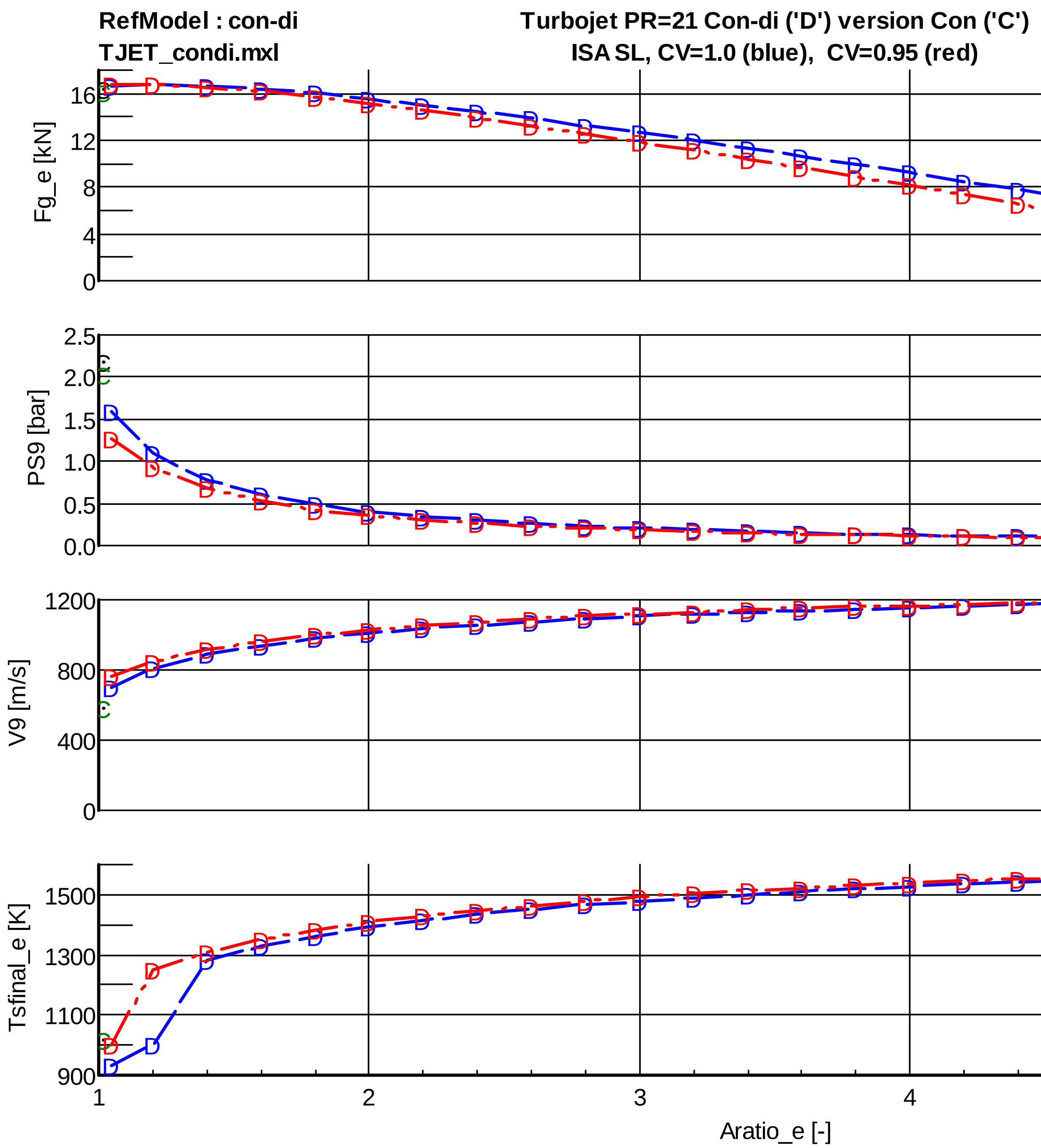
A_geom = CD * A_eff

The effective area is lower than geometric area due to boundary layer effects on the flow (reduced velocity due to friction).
Either design nozzle exit A_eff or A_geom can be user specified using a user specified CD, or directly user specified depending on selected Exhaust nozzle options.

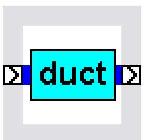
The graph below shows the 3 different operating regimes of a con-di nozzle of a turbojet example (sample project TJET_condi.mxl) model. Nozzle pressure ratio is 4 and ambient conditions are ISA SL:

- con-di ratio 1 - 1.155
  Here the nozzle is underexpanded up to 1.155 where the nozzle is fully expanded and maximum gross thrust is obtained. There are only oblique shocks downstream outside the nozzle.
- con-di ratio 1.155 - ~4.7
  Here the nozzle is overexpanded with the shock outside the divergent nozzle
- con-di ratio ~4.7 -
  Here the nozzle is overexpanded with the shock inside the divergent nozzle.

With the `Ideal con-di nozzle complete expansion` option checked, shocks are ignored and ideal complete expansion to con-di nozzle exit is assumed, usually resulting in the maximum theoretical nozzle gross thrust FG. The effect of CV can be seen comparing the blue and red curves in the graph.



8.4.1.1.7 Duct



Use the duct component to model components just passing flow to the next component at the expense of an amount of [pressure loss](#) or when a user specified heat flux must be added to or taken from the gas. Note that the duct only provides a simple means to model [heat transfer](#) using a design and/or an off-design heat flux.

Often, secondary air flows from other components such as compressors 'leak' into the duct. To model this effect, bleed flows can be specified to enter the duct, using the table in the `Bleed in-flow` tab sheet. Use the bleed flow numbers to identify the appropriate compressor bleed flows, and specify the fractions of bleed flows actually entering the duct. In case the corrected entry mass flow is used for the pressure loss calculation, this flow does not include the bleed flows! By definition bleed flows bleed into the entry of the duct. If bleed in-flow functionality e.g. halfway the duct component is to be modeled, the modeler should use two ducts in series using the inflow of the second duct.

An unlimited number of duct bleed flows can be specified in the `Bleeds` tab sheet. Duct bleed flows are secondary air flows extracted from the duct air flow for various purposes: bleed flows can be used to model customer bleeds or turbine cooling bleeds used in the turbine component, etc. The bleed flow numbers used must be unique for the model.

8.4.1.1.8 Link Bar

The link bar component is used to establish gas path or control links between components which are positioned with a certain distance in between in order to enhance surveyability of the components on the model panel. The bar can be resized and rotated (right-click the component and select `Rotate` or press ALT + R for a selected model component).

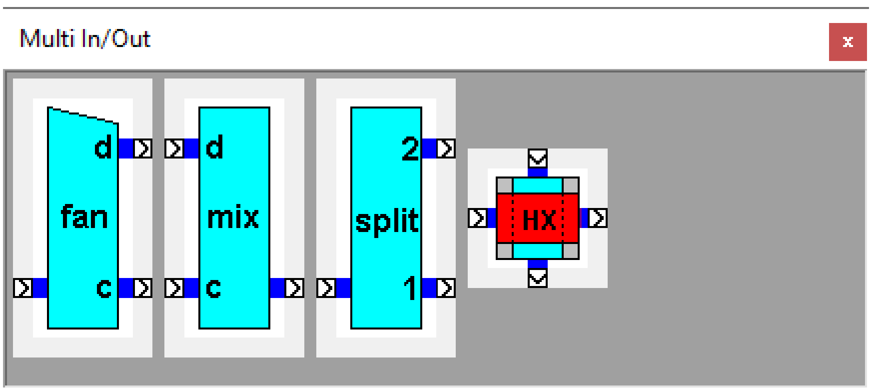
The colors of the link bar are drawn automatically when valid connections between the components have been made.

- Blue
  This link represents a primary gas flow passing from component 1 to component 2
- Black
  This link represents a control input from component 1 to component 2

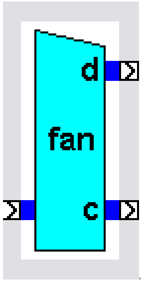
8.4.1.2 Multi in/out Component Library

The GSP Gas Path components includes the following gas path components:

- Fan
- Mixer
- Flow Splitter
- Heat exchanger/recuperator



8.4.1.2.1 Fan



The fan component is used to represent the fan or low pressure compressor in turbofan engines including the splitter dividing the fan exit flow in a duct- (or "bypass") and core flow. The compression process is modelled entirely equal to the compressor component, except that separate design data (pressure ratio and efficiency) and compressor maps for the duct- and core flows are used.

The fan has similar bleed flow functionality as the compressor component. The fan's bleed flow functionality applies to the core flow (no bypass bleed functionality).

The bypass ratio determines the ratio of the duct and core mass flows and is user specified in the design point (`Design bypass ratio`). The off-design bypass ratio usually deviates from the design value depending on the thermodynamic state of the gas turbine system (and is calculated as a state variable). In the design point, the duct flow is compressed using the duct design data and duct map; the core flow using the core design data and map. With deviating off-design bypass ratios, the "dividing streamline" between the two flows will tend to move, depending on effects such as the position of the splitter behind the fan. Since the core and duct maps represent geometrical parts of the fan, the distribution of the entry flow to the core and duct maps then may well have to change for best simulation accuracy. This change is controlled by the user specified Cf factor in the General tab sheet. The Cf factor ranges between 0 and 1:

- Cf = 0 implies the dividing streamline is not affected by the bypass-ratio at all and flow division between core and duct maps remains unaffected by deviating off-design bypass ratios. This case could well be thought of as a flow "splitter" at infinite distance from the fan exit.
- Cf = 1 represents the other extreme with the flow division between the two maps being entirely proportional to the bypass-ratio. This case seems hard to realise in practice, but a flow splitter right behind the fan would require a Cf at least larger than 0.
  In practice, best results are obtained with Cf values close to 0.

For both duct and core flow paths options are available off-design performance prediction. Using the radiogroup box one of the following options can be chosen:

- `Map`
  The standard option uses a map similar to the compressor map of type component map. The map file consists of tables with corrected mass flow, efficiency and pressure ratio as a function of corrected normalised rotational speed and beta. The map operating point corresponding to the design operating point is specified using the map design rotational speed and beta values. Click the `Graph` button in the `Map` tab sheet to view the map graphically and note the yellow rectangle which can be used to move the map design point. In the map graph also the beta-lines can be show after activating the appropriate item in the `Options` menu.
- `Operating Line`
  This advanced option allows the engine modeller to use the compressor operating line. In case the maps of the compressors cannot be obtained from the manufacturer, the operating line of an existing engine can inserted (or a predicted operating line for a new engine). When

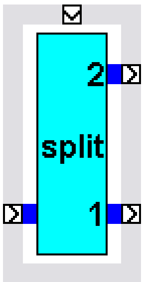
using this option note that the mass flow error equation for the turbine needs to be deactivated to obtain a solvable equation system. In case the operating line is the actual running line of the engine, the error will remain very small (can be visualised through the option Werror on the turbine output tab sheet). Using a predicted operating line, the error increases the further the operating point is located from the real operating point.

- `No map (DP only)`
  This advanced option allows the engine modeller to calculate design point performance without having to specify an off-design component map.

The option `Reset map scale factors` sets the map scaling factors to 1 during off-design calculations. This enables the engine modeller to match existing turbo components using their actual/original maps.

The `DownStream Calculation` option is to specify which downstream gas path (duct or core) is calculated first. This is for cases where bleeds must be calculated in one downstream gas path that is used in the other. For example: is fan duct bleed is inserted in core flow (e.g. behind LPT), the duct flow must first be calculated (DownStream Calculation = Duct first). In the case of both flows from duct to core and vice versa, the problem can not be made 'determinate' anymore and splitter and mixer components must be used (instead of bleeds/sec. airflows) to model the secondary flows, like in the ABFAN_HEM model.

8.4.1.2.2 Flow Splitter



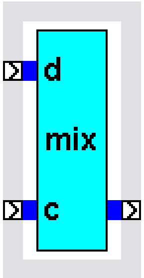
The flow splitter splits a flow into two flows according to the Split Fraction SR. SR is the fraction of Splitter entry flow that is diverted to the secondary exit. So with SR = 0, all is going to exit 1, with SR = 1 all is going to exit 2. With SR = 0.5 for example, each exit gets half of the flow.
The Split fraction can be `free state`, `user specified fixed` or `user specified variable`. The last option requires an additional Flow Splitter Control component.

Use the split fraction for blowing off air somewhere from the gas path, either to model large compressor bleed flows (APU case) or secondary air flows that may go into a gas path elwsewhere using a Mixer component.

Specify design split fraction and exit static conditions as `area`, `velocity` or `mach nr` on the `Design` tab sheet.

The `DownStream Calculation` option is to specify which downstream gas path (1 or 2) is calculated first. This is for cases where bleeds must be calculated in one downstream gas path that is used in the other. For example: is bleed from a duct downstream of exit 2 is inserted somewhere downstream exit 1, the 2 flow must first be calculated (DownStream Calculation = 2 First).
In the case of both flows from 1 to 2 and vice versa, the problem can not be made 'determinate' anymore and splitter and mixer components must be used (instead of bleeds/sec. airflows) to model the secondary flows, like in the ABFAN_SECAIR model (same case for Fan instead of Splitter).

8.4.1.2.3 Mixer



The mixer component is used to simulate the mixing of two gas flows to a single flow, such as the mixing of the bypass and core flows in a mixing turbofan engine. The mixing of the two flows to a uniform gas condition (i.e. temperature, composition) is assumed to completely occur in an infinitely short distance without losses. To incorporate pressure losses, use a preceding or downstream duct component. Mixer exit flow conditions are determined using the equation for conservation of momentum with the assumption that the ratio of entry flow static pressures remains constant. For applying this equation, entry and exit flow static conditions and velocities need to be calculated and therefore the flow cross-areas need to be specified. 5 options for specification are available:

- Specify both entry cross section areas for duct and core.
  In this case the static pressure ratio is determined in the design point calculation and maintained for off-design calculations.
- Specify total cross section area and static pressure ratio.
  In this case the duct and core cross section areas are determined from the static pressure ratio during the design point calculation.
- Specify core mixing plane entry parameter (depending on parameter selection) and static pressure ratio.
  In this case the duct and core cross section areas are determined from the static pressure ratio during the design point calculation.
- Specify duct mixing plane entry parameter (depending on parameter selection) and static pressure ratio.
  In this case the duct and core cross section areas are determined from the static pressure ratio during the design point calculation.
- Join 2 flows: no conservation of momentum is maintained, only conservation of energy joining the flows into one.

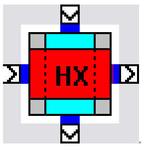
Important note:
At design point calculation the inflow areas are either specified or calculated from the gas inflow V or Mach. The duct inflow and core inflow areas or static conditions are determined by the mixer input options, and therefore do **NOT** correspond anymore to upstream connected components exit conditions! Mixing plane entries are to be considered connected to upstream via perfect duct with varying area (if upstream areas are specified).

The specification of this data often requires some trial-and-error cycles, especially if the area data is not known. In that case, use the second option for example and set static pressure ratio to 1. Repeat design point calculations until mixer entry Mach numbers are in a reasonable range (around 0.5). Illegal mixer parameter values include Mach numbers exceeding 1 and static entry pressures exceeding the total pressure of the other entry (this would cause reverse flow) and are reported as errors (see 4.7). The exit flow cross-area is assumed equal to the sum of the entry cross-areas.

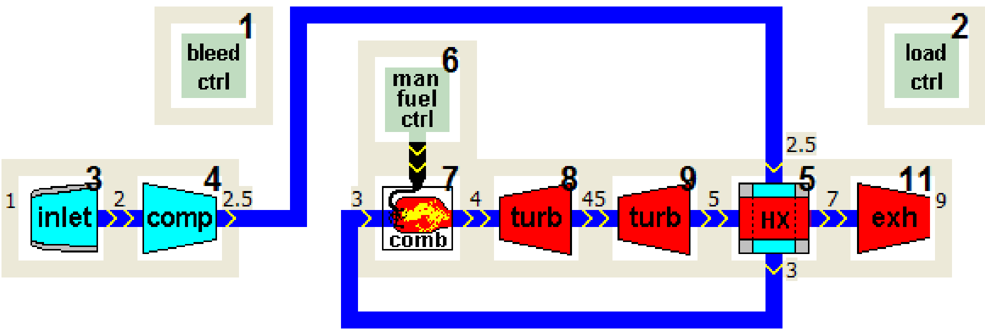
Static mixing plane conditions
The mixer calculates separate mixing plane static conditions, required to calculate the equations relating the two mixing flows (usually using conservation of momentum and assuming a user specified relation between the two entry flow static conditions).

Note that the mixing plane entry is assumed to be *between* component inlet and exit stations. These are NOT the mixer inlet station static conditions corresponding to the upstream component Exit static conditions that are calculated if non-zero upstream component exit area, mach or velocity values are specified. Since the mixing plane entry is not coinciding with an engine station, mixing plane entry mach, area and velocity output parameter ID's are Mach_mduct, Mach_mcore, A_mduct, A_mcore, V_mduct, V_mcore followed by component ID string.
The mixing plane exit is corresponding to the component exit conditions and therefore the exit station number, so mixing plane exit conditions are simply represented exit station conditions.

8.4.1.2.4 Heat exchanger/recuperator



The heat exchanger component is used to simulate heat transfer between two gas flows in a heat exchanger. In a gas turbine heat exchangers are often used as "recuperators", transferring heat from the exhaust gasses to the combustor entry air to improve cycle efficiency.
The two flow passages are identified by numbers 1 (hot flow) and 2 (cold flow). These numbers are also used to identify output parameters such as for example the pressure ratio in passage 2 in the figure below with PR2_4 (_4 indicating component 4).
For the design point, either design point heat flow rate or the temperature change in the passage first entered by the gas (i.e. calculation procedure) is specified. Be careful to enter the right signs for the design heat flow (positive for heat from passage 1 to 2).
In the *TSHAFTrecup* sample project in the figure below for example, the first passage is the number 2 passage receiving air from the compressor.



Off-design heat transfer is determined by a component map providing heat exchanger effectiveness as a function of passage 1 and 2 mass flow rates. As with other maps, the heat effectiveness map is automatically scaled to the engine design point heat flow and passage 1 and 2 mass flows using the map design point.

In GSP, the heat exchanger / recuperator component definition of effectiveness is fully enthalpy based:

$$Eff = Q / Qmax$$

with Q = the actual steady-state heat flow from flow 1 to flow 2 :

Q = Wh * (H(Th_in, GCh_in) - H(Th_out, GCh_out))
or Q = Wc * (H(Tc_out, GCc_out) - H(Tc_in, GCc_in))

and Q max = Wcpmin * (H(Th_in, GCcpmin) - H(Tc_in, GCcpmin))

where:

Wh is the hot side mass flow, Wc is the cold side mass flow,
H(T.. ,GC..) is the GSP enthalpy function of temperature (i.e. varying Cp) taking into account gas composition GCh (hot flow) and GCc (cold flow),
Th_in/out are the hot side in/out total temperatures
Tc_in/out are the cold side in/out total temperatures
Wcpmin is the mass flow of the flow with the lowest product W*Cpin (mass flow * entry value for Cp).
GCcpmin is the gas composition of the flow with the lowest product W*Cpin (mass flow * entry value for Cp).

Internal heat soakage effects (heat capacity of 'wall' material between two gas flows) can be modeled by specifying:

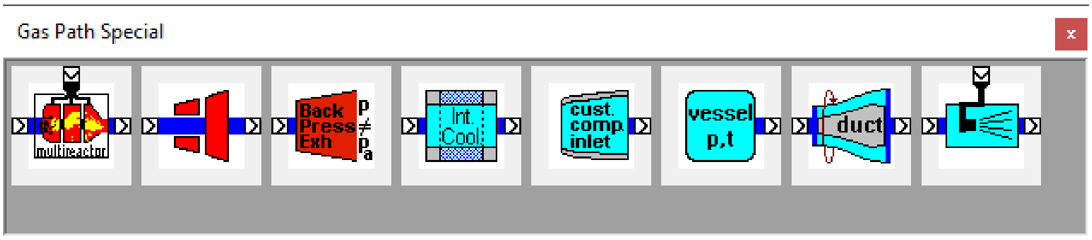
- effective contact surface [m^2]
- design film coefficient ratio
- wall effective mass [kg]
- wall material specific heat [J/kg/K]
- effective wall thickness [m]
- wall material thermal conductivity [W/m/K]
- average wall temperature time constant

For both passages volume effects can be calculated and pressure losses can be specified. Note that the heat soakage (transient) and heat sink (steady-state and transient) effects are applied to the 2 gas flows without taking into account the heat flux between them. In that sense the heat soakage effect can be considered as 'Outside wall heat soakage effect' and the heat sink heat transfer is simply happening between the flow (flow 1 or flow 2) and the specified Heat sink. For heat soakage effects of the material between the two gases, specify the data in the `Internal Heat soak` tab sheet.

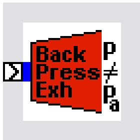
8.4.1.3 Special Gas Path Component Library

The Gas Path Special Component Library includes the following gas path components:

- Multi Reactor Combustor
- Turbine Stage
- Back Pressure Exhaust
- Inter Cooler
- Custom Composition Inlet
- Pressure Vessel
- Rotating Duct
- Fuel pre-Mixer



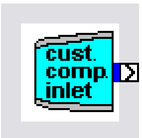
8.4.1.3.1 Back Pressure Exhaust



This exhaust inherits functionality from the generic Exhaust Nozzle component in the Gas Path Component Library; refer to these help items for more information.

This fixed area exhaust component can be used to specify the ambient pressure at the exit of the exhaust (different than the ambient/flight conditions) to be typically used for industrial applications to simply simulate pressure losses in the boiler behind the exhaust component.

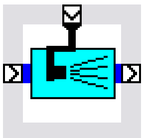
8.4.1.3.2 Custom Composition Inlet



This inlet component is a derivative of the Inlet of the Gas Path Component Library. Inheriting all inlet functionalities this component is extended with the ability to specify a different gas composition entering the component which overrides the standard gas composition. The standard gas composition (air) is initially defined when a new component has been dropped on the model panel.

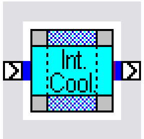
Design and off-design compositions can be defined in their respective tab sheets.

8.4.1.3.3 Fuel pre-Mixer



Pre-mixer/pre-vaporizing fuel component. Fuel is added in a duct similar to a combustor component. The difference is that no combustion takes place. The input window is quite similar to the combustor component. There is no `General` tab sheet, and the `Design` tab sheet has functionality to select the design fuel in terms of mass flow, or as FAR. Since there is no combustion taking place, only mixing, efficiencies related to the combustion process (e.g. maps) cannot be used.

8.4.1.3.4 Inter Cooler



Cools the gas turbine's flow using water or specify a custom Cp for other fluids as cooling flow.

`General`

`External control of coolant flow`
Enable or disable an external coolant flow controller

`Coolant control input`
Input values for both coolant mass flow and coolant temperature

`Off-design effectiveness`
Fixed effectiveness or effectiveness from map

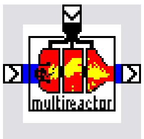
`Design`
`Specify design HX operating point`
Specify the design of the HX using `Heat flux`, `1st pass flow temp. change`, `1st pass flow exit temperature`, or `Effectiveness`.

`Coolant`
Design input values for both coolant mass flow and coolant temperature. Specify water or a user defined coolant by entering a custom Cp.

`Effectiveness map`
Define map and heat capacities

`Internal heat sink`
Define the properties of the internal material of the heat sink (i.e. the material in between the two flows).

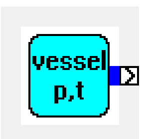
8.4.1.3.5 Multi Reactor Combustor



This combustor is an child component of the Combustor of the Gas Path Component Library, inheriting all functionality. This type of combustor allows modelers to more accurately describe the combustor geometry in calculating emissions. Combustor compartments (sections) can be added to model the actual global processes that describe the forming of emissions more accurately.

This extra functionality can be found on the `Emissions` tab sheet where an additional emission model has been added to the option list. Upon selecting the emission model a `Multi-Reactor` tab sheet, with input options to define the combustor compartments, appears.

8.4.1.3.6 Pressure Vessel

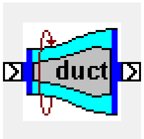


Specifies constant pressure, temperature and (design) mass flow (vessel of immeasurable size).

The `General` tab sheet `Model Options` can be used to either deliver a constant vast amount of air of specific temperature and pressure for design and off-design use (option `Constant user specified pressure and temperature`) or an amount specified by an external pressure and temperature controller (`Use pressure and temperature control component`) to control the off-design temperature and pressure.

Note that the off-design mass flow is a state variable, similar to the inlet. To specify a certain off-design mass flow, a nozzle (e.g. accompanied by a nozzle controller and an equation controller) needs to be added to the model.

8.4.1.3.7 Rotating Duct



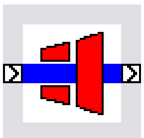
The input is basically identical as for the Duct, since this duct rotates additional input can to be given for:

`Rotation data` on the `Design` tab sheet

- `shaft nr./suffix`
  Specify the shaft that drives this duct.
- `Radius in`
  Specify the mean radius of entry
- `Radius out`
  Specify the mean radius of exit

The difference between entry and exit velocity is a measure for the kinetic energy increase/decrease of the fluid.This difference will be added to the enthalpy from which the output conditions will be calculated.

8.4.1.3.8 Turbine Stage



Inheriting basic functionality from the turbine details are added to model a single turbine stage.

The gas conditions of the NGV exit will be calculated accordingly to the specified input. The basic functionality is similar to the turbine. Calculations are based on a chocked nozzle guide vane. Pressure losses due to friction (non-isentropic expansion) are taken into account. To calculate the pressure losses, the isentropic (ideal) expansion pressure is compared to the non-isentropic expansion pressure corrected for $\eta_{is,turbine}$. This pressure loss can be distributed over the nozzle and rotor using the `Stator loss fraction` numeric input field. Default the pressure loss is distributed equally (0.5) over the nozzle and rotor blade rows. In case the nozzle is responsible for the entire pressure loss, the `Stator loss fraction` needs to be set to 0, if the rotor is responsible for the entire pressure loss, the fraction needs to be 1.

8.4.2 Engine Control Components

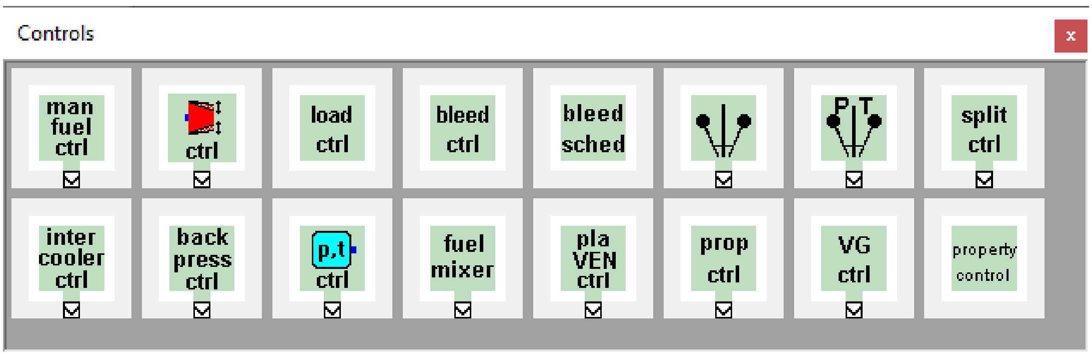
The engine control components (or "Controls") represents components that directly control engine inputs such as fuel flows, variable geometry, bleeds and other operating conditions and are arranged the following libraries:

- Standard Controls Component Library
  Contains the standard control components.
- Power Control Component Library
  Contains power control components.

8.4.2.1 Standard Controls Component Library

The Control Library contains generic control system component models. Note that these controls all apply to OD simulation mode only.

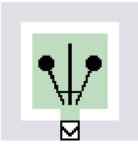
- Manual Fuel Flow control
- Manual Variable Exhaust Nozzle control
- Power Turbine Load control
- Compressor Bleed control
- Bleed Schedule Control
- Compressor Shaft Speed Governor
- Turboshaft Governor Fuel control
- Flow Splitter Control
- Inter Cooler Control
- Back Pressure Exhaust Control
- Pressure Vessel Control
- Fuel Mixer
- Variable Exhaust Nozlle Control (PLA controlled)
- Propeller Control
- Variable Geometry Control
- Property Control



8.4.2.1.1 Compressor Bleed control

The compressor bleed control offers flexible control of both design and off-design compressor bleed flows. Use this component for specifying variable bleed (e.g. for analysis of bleed flow effects) instead of the bleed flow specification in the compressor component that is meant for specification of constant compressor bleed flows or bleed flow fractions only. Any type of compressor bleed flow can be controlled (e.g. customer bleed, handling bleed, turbine cooling bleed flows etc.).
Either absolute bleed mass flow or relative bleed mass flow (relative to total compressor entry mass flow) can be specified (), if desired a function of time in the `transient` input tab sheet.

8.4.2.1.2 Gas Generator Fuel control



The gas generator fuel control component offers a generic fuel control system model with PID speed control and maximum acceleration and minimum deceleration schedules. Many simple conventional (mechanical) gas generator rotor speed governors can be modeled accurately. For more complex (e.g. FADEC, Full Authority Digital Engine Control) control systems characteristics, the component is useful when detailed control aspects can be neglected, such as in gas turbine transient performance trade-off studies. For detailed models of complex control systems, usually custom components are required. Also for accurate transient simulation of modern multi-spool jet engine (e.g. turbofan engines) control systems, controlling both gas generator and fan rotor speeds, custom components are required.

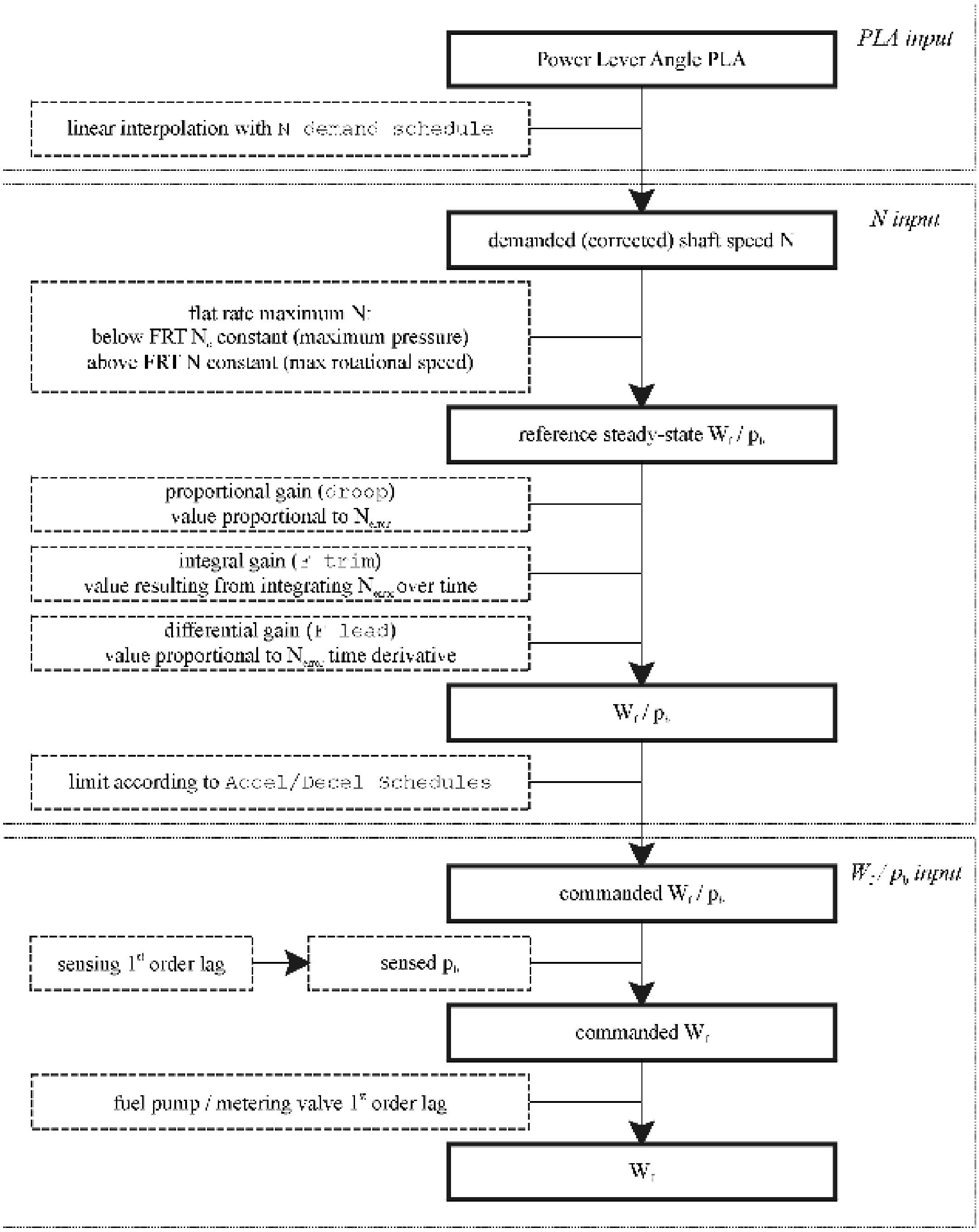
Schedules
Three schedule tables must be specified:
1) a PLA - Ndem schedule representing the relation between power lever angle PLA and demanded gas generator rotor speed (or corrected compressor rotor speed),
2) governor PID gain schedules including a reference value (WfPbref) al as functions of rotor speed (or corrected rotor speed),
3) Maximum acceleration and deceleration WfPb schedules as functions of rotor speed (or corrected rotor speed).

Click the `Graph` button to see graphical presentations of the schedules. All schedules are represented by tables using linear interpolation. Note that outside the table range, values are extrapolated !

Scaling
All schedules are scaled to design point values. From the PLA design (in the `Design` tab sheet) the design Ndem is determined and scaled to the design rotor speeds specified for the compressor component. The scaling factor is saved and used for scaling between the actual Ndem table values and actual compressor speed. Also scaling factors are determined for the Wf/Pbref values in the PID schedule and the WfPbmin/max values in the maximum accel/decel schedules. Note that control characteristics (i.e. control components) from one engine can easily be used for another one, but simulation results (for WfPb for example) may deviate significantly from the schedule table values.

Control system input
There are up to 4 controller input parameters :
- Power Lever Angle (PLA),
- gas generator compressor entry temperature,
- total pressure of a user specified compressor bleed flow to represent burner pressure
- rotor speed.
  Sensor dynamics for rotor speed, pressure and temperature are modeled assuming first order lag with user specified time constants.

The calculation of fuel flow Wf is performed according the chart below.



The controller governs a demanded rotor speed level, which is a function of PLA. Depending on the `Corrected Ndem` option in the `Schedules` tab sheet, the demanded rotor speed schedule represents either actual (Ndem) or compressor entry corrected rotor speed (Ncdem=Ndem/√θ).
Demanded rotor speed Ndem (or Ncdem) is controlled using a PID controller with P(proportional), I(integral) and D(differential) gains depending on scheduled rotor speed Nsch. Nsch may either be Ndem or Ncdem, depending on the user specified option `Corrected N in Gov. and A/D sched` option in the `Schedules` tab sheet.

The PID controller outputs a Wf/Pb signal representing fuel flow divided by burner pressure. The PID output is based on the rotor speed error (Nerror = Nsch - N) and is added to the reference WfPb (Wf/Pbref), which is a function of Ndem.
The integrator in the PID controller is a reset integrator, resetting the integrator signal to zero upon input signal sign change. This may not apply to all PID controllers, for other PID logic, custom control components are required.
As an option, Wf/Pb corrected to compressor entry temperature may be used instead (i.e. Wf/(Pb.√q) depending on the `Correct Wf/Pb to Ttin compressor` option in the `Schedules` tab sheet.
The WfPb (or corrected Wf/Pb) signal is limited by the maximum acceleration or deceleration schedule, which are user specified functions of actual rotor speed or corrected rotor speed, depending on the user specified option `Corrected N in Gov. and A/D sched` option in the `Schedules` tab sheet.
From the resulting limited Wf/Pb, commanded fuel flow Wfcalc is calculated by multiplication with Pb (and for corrected Wf/Pb also with √q).
Actual fuel flow Wf is calculated using a first order lag transfer function with a user specified time constant to simulate fuel pump dynamic response.

Automatic flat rating control can be used to limit power setting, based on maintaining constant (maximum) corrected rotor speed at outside air (or compressor entry) temperatures below the flat rated temperature FRT. The maximum corrected rotor speed is defined the maximum rotor speed at FRT.

In the chart, the following remarks apply:

- Ndem is corrected for the compressor entry temperature if `Corrected Ndem` is checked
- Flat rating is only applicable if `Auto Flat Rated Power Limiting` is checked
- Nerror is the difference between N and demanded N

Other options:

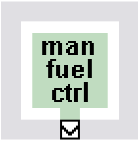
- `Fully Trimmed steady state (no droop)`
  Check this option if fully trimmed steady state operating points (proportional gas generator speed governor droop error fully compensated by trimmer) need to be calculated. This enables calculation of operating points at user or control system specified rotor speeds and shaft loads. With this option, GSP calculates the final integrator output signal at the fully trimmed steady state condition (i.e. at infinite time).
  Both a state and an error variable are added with the 'Fully trimmed steady state' option: the trimmer output then is a state and the shaft speed error (Ndemand-N) is the error. No additional component needs to be modified and fully steady state points with a fully trimmed gas generator speed can be calculated.

Note that this only works if the integral (I) gains in the PID controller are non-zero and that, with this option, transient simulations are inhibited !

- `F trim reset to 0 Nerror dead band`
  This option is to control the dead band for the integrator reset upon sign change of the integrator input signal (i.e. Nerror). Integrator reset only occurs after Nerror has passed zero with the specified value.

Note: steady state and steady state series calculations may become problematic when control system components with engine parameter feedback depending on time (such as Gas Generator Fuel control and Turboshaft Fuel control components) are used in the model. The best way to calculate a steady state point then is to stabilize at the particular operating point using a transient calculation first before performing a steady state calculation.

8.4.2.1.3 Manual Fuel Flow Control



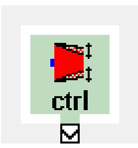
The manual fuel flow control component enables direct specification of off-design fuel flow to a combustor component (design fuel flow is specified in the combustor component). This component is the simplest for specifying power setting of a gas turbine engine model, disregarding any control system logic. Usually the manual fuel flow control is used for steady state off-design engine performance calculations, either single off-design points or steady state series parameter sweeps using the input table (numeric grid) to define a range of fuel flows.

The manual fuel flow control can also be used to calculate transient responses on fuel flow variations. An example would be the calculation on fuel flow step responses for control system design (using system identification techniques?). The fuel flow step functions then are specified in the transient input table.
The manual fuel control `Specify` option settings are similar to those in the combustor component linked to it. However, the selected option in the combustor may be different from that in the manual fuel flow control, to facilitate different ways to specify power setting DP and OD.
Note that
Note that upon selecting SOT input, an equation needs to be added, so a reset of the model and recalculation of the DP is required. SOT is evaluated in an error equation (with Wf as state) in the iteration towards user specified SOT (iteration necessary because SOT may be affected by downstream HPT cooling flows).

Fuel flow also can be specified as a 'free state' in order to calculate an off-design operating point with (instead of user specified fuel flow Wf, combustor exit temp. or fuel-air ratio) an alternative user specified power setting condition such as turbine rotor speed and/or power load ('Power balance at rotor speed' turbine component model option). Using the Fuel flow as a free state model option always requires the setting of another component's option to provide an extra error variable to maintain an equal number of model state and error variables. If this requirement is not met (no corresponding settings in two components), a 'Model configuration inconsistent' (Nstates<>Nerrors) error is reported.

The checkbox `Always create state` specifies whether the state is added to the equation system. This advanced option is usually set by other components when configuring the model.

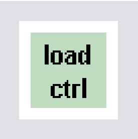
8.4.2.1.4 Manual Variable Exhaust Nozzle Control



The manual variable exhaust nozzle control component enables direct specification of off-design exhaust nozzle areas (design area is calculated during the design calculation). Throat area and, depending on the convergent-divergent (`Con-Di`) option setting in the `General` tab sheet, also exit area, are specified relative to the design point values (i.e. "1" is the design value). This component offers the simplest way of specifying nozzle area, disregarding any control system logic. Usually the manual variable exhaust nozzle control is used to perform steady state calculations with varying nozzle areas.

Note that in order to control the areas of an exhaust nozzle component, the `Variable Area nozzle` option in the exhaust nozzle component needs to be activated.

8.4.2.1.5 Shaft Load control

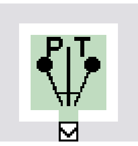


The shaft load control component offers flexible control of both design and off-design specification of loads on any shaft in a GSP model. Use this component for specifying (usually power) turbine loads instead of the external load specifications in the turbine component which are meant for specification of constant and relatively small power off-take (PTO) auxiliary loads.
The loads can be specified in terms of torque and/or power (if both are specified, the loads are added together) and as a function of time in the `transient` input tab sheet. This component is especially useful to specify time dependent load variations (torque steps) on power turbines.

Power/torque load is the specified power/torque that is extracted from the shaft the Load component is connected to. If correctly configured (PWsurplus=0) than this of course corresponds to PWshaft (assuming single turbine on shaft) for steady-state. For transients, a power delta remains causing accels or decels.
Total shaft Power (and Torque) outputs (PW_total and TRQ_total) are calculated from the combined user specified power and torque (PW_spec, TRQ_spec). For example, if only a torque load is specified (as TRQ_spec input), power load (input) is 0 then PW_total is the power calculated from the torque (trq x rpm*60/2pi). If both power and torque are specified, the PW_total is calculated by "adding" the power and torque according to: user spec. power + user spec. torque * rpm*60/2pi. TRQ_total then is calculated by adding the torque converted from the PW_spec to the TRQ_spec.

*Note that when using variable load in the transient input table to act upon a **free** power turbine, the free power turbine speed only changes during transient simulation ! For steady state calculations the turbine speed should be user specified (see the turbine component).*

8.4.2.1.6 Turboshaft Fuel control



The turboshaft fuel control component offers a generic fuel control system model for free power turbine turboshaft engines. It is derived (inherited) from the Gas generator fuel control component and includes all functionality to control the gas generator. In addition, the turboshaft fuel control has a PID controller (PT governor) for the power turbine shaft speed, which outputs a signal to the gas generator controlling either one of the following parameters depending on the `PT Governor output` option in the `PT Control` tab sheet:

- Power Lever Angle PLA
- Demanded gasgenerator shaft speed N
- Fuel massflow over combustor pressure Wf/pb

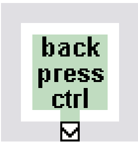
- Fuel massflow Wf

The gas generator control logic is fully inherited from the Gas generator fuel control component (including maximum acceleration / minimum deceleration schedules etc.). Functionality from the gas generator control component is used, with the starting point depending on the on the `PT governor output` option. In case the PT Governor outputs a Wf signal (option 4) for example, all Gas generator functions are skipped except for the fuel pump first order lag transfer function.
As with the Gas generator fuel control component, many simple turboshaft rotor speed governors can be modelled accurately. For more complex (e.g. FADEC, Full Authority Digital Engine Control) control systems characteristics, the component is useful when detailed control aspects can be neglected, such as in gas turbine transient performance trade-off studies. For detailed models of complex control systems, usually custom components are required.

Similar to the integrated Gas generator control, there is an additional 'Fully trimmed steady state' option (in the "PT control" tab sheet) for PT governor to calculate fully trimmed steady state operation with the power turbine power balance fully matched at user specified rotor speed. For the Power turbine, this option works different from the gas generator case: only a state is added for the the trimmer output. No error variable is added and an error variable has to be provided by another component. This will normally be from a (power) turbine component, i.e. a turbine component with the Power turbine option set and the 'Power balance at rotor speed' option set to add the error variable. This enables the calculation of fully trimmed free-power turbine turboshaft steady state points (the free power turbine power matching the power load).Normally the load would be varied and the 'Fully trimmed steady state' would provide fully trimmed control states providing sufficient fuel to maintain free power turbine rotor speed at this load.
Please note that this option only works for steady state calculations !

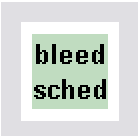
Note: steady state and steady state series calculations may become problematic when control system components with engine parameter feedback depending on time (such as Gas Generator Fuel control and Turboshaft Fuel control components) are used in the model. The best way to calculate a steady state point then is to stabilize at the particular operating point using a transient calculation first before performing a steady state calculation.

8.4.2.1.7 Back Pressure Exhaust Control



This controls the ambient pressure of the Back Pressure Exhaust component for OD steady-state (series) or transient calculations.

8.4.2.1.8 Bleed Schedule Control



The bleed scheduler offers flexible control for off-design bleed control using corrected rotor speed schedules. Use this component for specifying variable bleed (e.g. for analysis of bleed flow effects) based on a corrected spool speed using a predefined schedule. Any type of compressor bleed flow can be controlled (e.g. customer bleed, handling bleed, turbine cooling bleed flows etc.).

Either absolute bleed mass flow or relative bleed mass flow (relative to total compressor entry mass flow) can be scheduled as function of the corrected rotor speed.
Corrected rotor speed is either

- Shaft Nc (1st component entry conditions)
  The shaft corrected rotor speed is taken for the schedule input Nc. This is the rotor speed of she specified shaft corrected to ISA from the entry conditions of the first component in the gas path on the shaft, usually the compressor that the bleed is coming from. However, depending on your component and shaft configuration, this is not necessarily the corrected shaft speed of the compressor delivering the bleed, as would for example be the case if there is an upstream separate compressor on the same shaft.
- Corrected for ambient total conditions
  The rotor speed of she specified shaft is corrected to ISA from the total (for aircraft 'ram') conditions in the undisturbed flow upstream the engine.

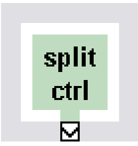
Note that the schedule must be descending and the grid does not extrapolate. Outside the input range the output value stays at the level closest within the schedule.

The mass flow scheduled value can also be specified as

- uncorrected (default)
- corrected to the component entry conditions,
- corrected to total (for aircraft 'ram') conditions in the undisturbed flow upstream the engine

If corrected, the mass flow resulting from the schedule is 'uncorrected' in order to determine the actual mass flow to be taken from the bleed..

8.4.2.1.9 Flow Splitter Control

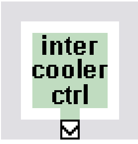


Controls the split fraction of an attached Flow Splitter component for steady-state (series) and transient analyses. The split fraction, or SR, is the fraction of Splitter entry flow that is diverted to the secondary exit. With SR = 0.5 for example, each exit gets half of the flow.

8.4.2.1.10 Fuel Mixer

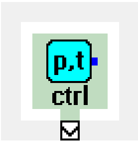
Specify a new custom (design) fuel to be a mixture of 2 different kinds of fuels. Note that this component only mixes 2 different kinds of `custom composition` fuels. A fuel pre-mixer of `standard` or `custom composition` fuels is the Fuel pre-Mixer. For both design and off-design, tab sheets are present to input the fuel compositions for the mixing fuels.

8.4.2.1.11 Inter Cooler Control



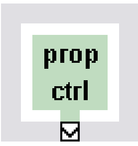
Control the mass flow and inlet temperature of the coolant flow for of the Inter Cooler component.

8.4.2.1.12 Pressure Vessel Control



In the series tab sheet a series of `Pressure` and `Temperature` as function of time/point values can be specified to be used in transient/steady state series calculations.

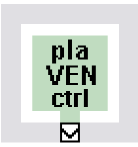
8.4.2.1.13 Propeller Control



Controller for the propeller component to set propeller mode to a constant speed propeller.

The control specifies the propeller Beta (angle of incidence of propeller blade).

8.4.2.1.14 Variable Exhaust Nozzle Control (PLA controlled)



This variable convergent exhaust nozzle schedules the relative throat area to a Power Lever Angle (PLA)setting.

Both the `ENP Ref(erence)` the `Nozzle rel(ative) Area` schedule must be ascending. Note that this component uses a first order equation for the calculation of the ENP (Exhaust Nozzle Pressure), for which opening and closing rates can be defined.

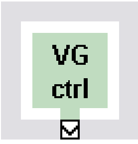
`ENP rate opening`
F trim reset to 0 after EPR error crosses 0 +/- dead band

`ENP rate closing`

`ENP rate time const.`

`ENP free play`

8.4.2.1.15 Variable Geometry Control



This control can be used to externally control the variable geometry (VG) of compressors and turbines.

8.4.2.1.16 Property Control

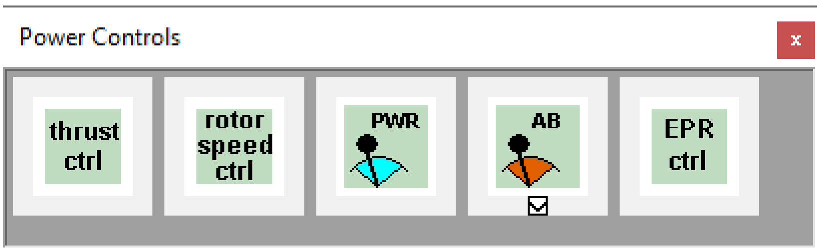
The Property Control component provides direct OD control over GSP properties and is for advanced users only. During simulation the property parameter value is simply set to the specified value, ignoring interference with any other processes such as iterations or model control schedules. This could possibly interfere with (thermodynamic) relations or equations!

This component not only enables the direct setting of component properties, but can also make free state variables of these properties when the `Control input value` option is set to `Free state`. This latter option enables users to turn any property into a variable; note that adding an extra free state variable into the equation set requires the addition of an error equation (e.g. an equation schedule component).

8.4.2.2 Power Control Component Library

The Power Control Component Library contains generic control system component models:

- Thrust Control
- Rotor Speed Control
- EPR Control
- Power Controller
- Afterburner Control



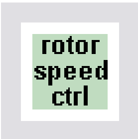
8.4.2.2.1 Thrust Control

Use this component to specify thrust for a model.

This component adds an error equation to the model (thrust - requested thrust) and thus requires to set an extra state to the model (e.g. fuel flow to free state).

NOTE:
This component is deprecated and does not work with components from the Case Control Component library. The functionality is replaced by the equation schedule components. The component is kept for backwards compatibility.

8.4.2.2.2 Rotor Speed Control



Use this component to specify rotorspeed of a selected shaft.

This component adds an error equation to the model (spool speed - requested spool speed) and thus requires to set an extra state to the model (e.g. fuel flow to free state).

Specify a shaft which needs to be controlled and choose a unit for the controller parameter; RPM (default), %, or corrected %.

NOTE:
This component is deprecated and does not work with components from the Case Control Component library. The functionality is replaced by the equation schedule components. The component is kept for backwards compatibility.

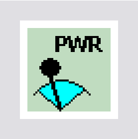
8.4.2.2.3 EPR Control

Use this component to specify the Engine Pressure Ratio (EPR). The pressures used for determining the EPR value can be chosen from the 2 drop down list boxes, usually the ratio of PT7/PT2 is used by low bypass turbofan engines, and for high bypass turbofan engines e.g. PT45/PT2 (see the aerospace standard 755 for station numbering). Some engine manufacturers use the total exit pressure of the bypass; PT14/PT2 (station 14 is located in the bypass channel between the fan exit, 13, and the nozzle inlet, 16 or 17).

This component adds an error equation to the model (EPR - requested EPR) and thus requires to set an extra state to the model (e.g. fuel flow to free state, or nozzle area).

NOTE:
This component is deprecated and does not work with components from the Case Control Component library. The functionality is replaced by the equation schedule components. The component is kept for backwards compatibility.

8.4.2.2.4 Power Controller



The Power Controller control component can be used to specify how the engine is controlled. This components links a parameter like a Power Code or a Rating Code to an engine power setting through a user defined scheduled engine parameter. A developed standard for the designation of gas turbine engine power settings is described by the SAE in Aerospace Standard 681.

The `General` and `Design` tab sheets specify the off-design and design conditions for the power controller respectively. A `Power Mode` and the `Power code` value can be specified for both design and off-design operation. The power modes are defined on the `Schedule` tab sheet.

The `Schedule` tab sheet allows the specification of the schedules. The `Scheduled Parameter` dropdown list allows the selection of the parameter that is scheduled (e.g. a rotor speed or EGT parameter). If a map is used, also the X and, in case of a 2-D map, Y input parameters for the map must be selected. In the table, columns are defined for `Power code` and one or more power/operating modes. Different schedules can be specified for different modes (e.g. a Normal and Emergency power mode).

The `Power code` column must be filled with the desired power code values and the schedules must be constructed on **either ascending or descending `Power code` value**. Rows can be added using the numerical grid navigator directly above the top left of the power code schedule. The buttons with a triangle bitmap can be used to scroll through the rows, the button with an "A" adds a row, the button with a "+" inserts a row before the selection, the button with a "-" deletes the selected row and the button with the "X" clears all the rows. A popup menu appears when the power code schedule is right-clicked from which the following options can be selected:

- `Edit Mode Description`
  A textbox dialog form opens in which the user can define the new title for the selected column.
- `Add Mode`
  A Mode column is added after the last mode column.
- `Delete Mode`
  The selected Mode column is deleted.

The scheduled parameter can also be obtained from a map. In that case the user selects the schedule parameter cell next to the corresponding power code and presses the Browse button to select the map. The X (and Y in case of a 2-D map) input parameters for the map must be selected from the corresponding dropdown lists.

The scheduled power parameter value is scaled to the design value of the scheduled parameter. This ensures that an OD calculation following a DP, with a Power code equal to the Design Power, produces an OD point equal to DP.
The scheduled value is obtained by interpolation with power code in the selected Mode column. The values in the cells may be either numeric strings or map names, controlled by the Map control frame at the bottom. Note that the Map input and output itself are not scaled, only the resulting scheduled value is scaled.

In the (transient) series tab sheet a series of `Power codes` as function of time/point values can be specified to be used in transient/steady state series calculations.
The parameters power code and the scheduled value can be added to the GSP data output parameter list by checking the respective checkboxes on the Output tab sheet. Remarks can be entered in the memo field on the Remarks tab sheet.

8.4.2.2.4.1 Aerospace Standard 681

The power setting of the engine is usually defined by a Power Lever Angle (PLA). Other forms of power setting control can be either Power Code (PC) or Rating Code (RC). Since the PLA setting is sometimes unknown, the power setting can be best described unambiguously by defining standard settings to specific values.

The SAE (Society of Automotive Engineers) has developed a standard (Aerospace Standard 681) for the designation of gas turbine engine power settings.

When the PLA power setting values are unknown, the power setting can be replaced by either a Power Code or a Rating code, where the latter is a more specific setting for the power setting as is may have settings related to the flight/ambient conditions.

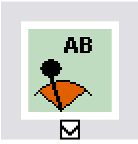
An example for the Power Code power setting is the following:

	Power Code	**Definition**
Augmented	100.0	Maximum
	to	to
	60.0	Minimum
Nonaugmented	50.0	Maximum
	to	to
	20.0	Minimum
Reverse	15.0	Maximum
	to	to
	5.0	Minimum

An example for the Rating Code power setting is the following:

	Power Code	**Definition Military**	**Definition Commercial**
Augmented	100.0	Maximum	Emergency
	90.0	Maximum Continuous	Maximum
	60.0	Minimum Augmented	Minimum
	55.0	-	Wet Takeoff
Nonaugmented	55.0	Maximum	-
	50.0	Intermediate	Dry takeoff
	45.0	Maximum Continuous	Maximum Continuous
	40.0	-	Maximum Climb
	35.0	-	Maximum Cruise
	21.0	Idle (Flight)	Flight Idle
	20.0	Idle (Ground)	Ground Idle
Reverse	15.0	Idle	Idle
	5.0	Maximum Reverse	Maximum

8.4.2.2.5 Afterburner Control



The simple AB fuel control component in this library has been developed to control the combustor component when used as afterburner. Afterburner power setting can be specified as either AB power code or directly by specifying `fuel flow`, `FAR`, `Texit` or `fraction of maximum stoichiometric fuel flow`.

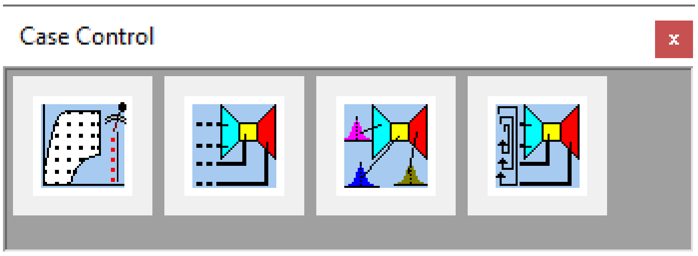
With AB power code, corresponding values for minimum and maximum AB fuel flow must be specified:

- With the `Fuel flow`, `FAR` or `Texit` options specified, minimum and maximum Fuel flow, FAR or Texit must be specified corresponding to minimum and maximum specified AB power codes.
- With the `Fraction of maximum stoichiometric fuel flow` option specified
  - a minimum fuel flow corresponding with minimum AB power code must be specified,
  - a value for the fraction of gas entering the afterburner that is not used for combustion (e.g. AB liner cooling flow) must be specified to determine maximum AB fuel flow.

At maximum AB power code, a fuel flow is applied that will realize stoichiometric combustion with the gas entering the afterburner minus the specified Gas fraction not used for combustion.

8.4.3 Case Control Component library

Case control components enable the predefining or scheduling of series of simulation inputs in various combinations. As such they provide a means to entirely predefine all simulation inputs requiring the user to only select the case and click the Run button the generate all output for the particular case. As an example, the entire engine operating envelope in terms of ambient/flight conditions and power settings or rating/power codes can be predefined and the corresponding performance calculated.



The Case Components Library includes the following components:

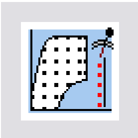
- Operating Envelope Scheduler (legacy dedicated flight envelope scheduler)
- Manual Case Control
- Loop Case Control
- Monte Carlo Case Control

The Operating Envelope Scheduler is a specific case control component to create input for a whole flight envelope, whereas the Manual Case Control and Loop Case Control components are generic.The Manual Case Control and Loop Case Control components are able to virtually change / set any variable in the model components, provided that the input is a numeric input value and active by option. This means that the user is not restricted in any way in using parameters that are not addressed by the transient tables. E.g. an off-design sweep for the ambient condition humidity is now possible.

The generic Manual and Loop Case Control components will adapt to selected case type in the project tree, effectively producing a means to centrally input case data for Design Point, Design Series, Steady State Point, Steady State Series or Transient.

Some case controller properties:
- Case controllers (except for the Operating Envelope Scheduler) can control equation controllers in a way that they 'override' the expression result (i.e. equation value to match the controlled parameter).
- With a case control component present in the model configuration,Transient/Series tab sheet will be hidden.
- These case components are the only components that can be deleted from the model configuration even if they have a parent. This is to ensure that a child case of a case can be setup with a different input controller.

8.4.3.1 Operating Envelop Scheduler



The Operating Envelope Scheduler can be used to create sweeps for sweeps of Flight/Ambient Conditions for a specific power setting sweep. This component adds Flight/Ambient Conditions in the models Ambient conditions transient table and a series of data points in the selected control's input transient table. Breaks are added in the Ambient Conditions transient table to define the last row of a sweep of a single parameter.

On the `Flight/Ambient Conditions` tab sheet the user defines the altitude, dTs (if ISA+ is chosen on the models off-design tab sheet of the `Flight/Ambient Conditions` form) and the speed of the aircraft/engine using the Start, End and Inc(rement) numeric fields. Upon changing the start-, end- or increment values the sequences are changed and directly displayed in a list below the numeric input fields. The numerical grids can be edited using the following keys:
- Insert key - Insertion of a new row
- Delete key - Deletion of the active line
- Arrow keys - Navigate through the grid (note that if the active row is the last row, the "down arrow key" will add a new row to the grid)

Limits can be entered for airspeed, Vc, or dynamic head (defined as the difference between total and static pressure; Pt - Ps), and a minimum altitude. Note that compressible flow equations are used to calculate q and V. The limits cutoff the regions specified by the numerical grids. Note that points at the limits can be generated with option `Outputs on limit`.

On the `Power/Control Settings` tab sheet the user defines the control component and sets the control setting values for a sweep. Note that only control components that have transient input capability can be selected as this is used by the Operating envelope scheduler. This means that currently (with version 11.0.2.5) only the controls from the 'Controls' and 'Power' libraries can be selected. Normally typical power setting controls would be used such as
- Manual fuel control and derived components
- Thrust control
- Nozzle control
- Rotor speed control
- Generic Power control and custom derivatives
- Simple AB fuel control and derivatives

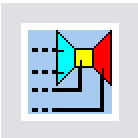
but also other controls can be used such as bleed controls to calculated bleed flow effects across the operating envelope.

On the `Options` tab sheet the user can select:

- `Auto-config model transient options`
  Specify the number of intervals between data points to decrease calculation stepsize and increase iteration stability.
- `Auto-deactivate non-looping transient inputs`
  Automatically deactivate all component transient inputs, to avoid interference with operating envelope schedule during steady-state series calculation.
- `Add Auto-breaks in table to separate data series`
  Add breaks in the `Flight/Ambient conditions` transient table to seperate data series so that the results can be plotted as individual sets.
- `Nr. of extra iteration intervals`
  Specify the number of intervals between data points to decrease calculation stepsize and increase iteration stability.
- `Start point nr.`
  Set to 1 to enable extra intervals for 1st point; leaving point 0 as unused start point in table; Note: only use either 0 or 1!

The buttons `Generate Envelope` and `Graph` generate and plot the flight envelope respectively.

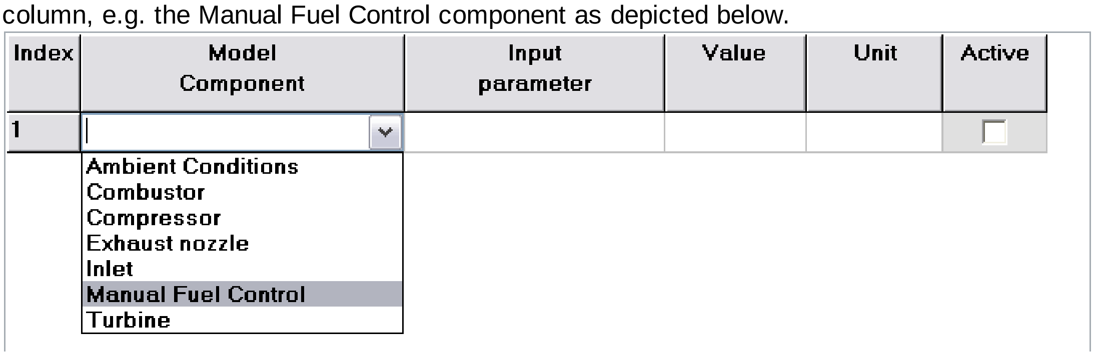
8.4.3.2 Manual Case Control



The Manual Case Control Component can be used as the central storage component for specific case input data. Effectively, the case input parameters in this component override the values set in the component data window.

In order to set the correct input for the model the modeler needs to set up the model so that the input that is required to be set by the Case Control Component is available in the component data window. If a parameter is not available it cannot be selected in the Case Control Component.

Single point case input (Design, Steady State case types)
To select case input data first the component needs to be selected from the `Model Component` column, e.g. the Manual Fuel Control component as depicted below.



Moving on to the next column the modeler can choose the `Input parameter` from a drop-down box. In this case the Manual Fuel Controls (which is an off-design case input component) input parameter Wf can be chosen since it is configured in its component data window, as depicted below. Upon selecting an active `Input parameter`, the `Value` and the `Unit` will be updated with the current value and unit of the numeric input field the `Input parameter` is referring to. Note thet the `Value` column is not available when the case mode is either Design Series, Steady State Series or Transient.

Index	Model Component	Input parameter	Value	Unit	Active
1	Manual Fuel Control				☐
		Wf			

Note that as long as the input is incomplete, inconsistent, or became obsolete by option in the component data window, the parameter will be shown in a red colored font. An example for an incomplete input parameter setup is shown above, an obsolete input parameter is shown below (in the component input window of the Manual Fuel Controller the input option changed from Wf to Texit).

Index	Model Component	Input parameter	Value	Unit	Active
1	Manual Fuel Control	Wf	0		☐

Series case input (Design Series, Steady State Series or Transient case types)
In case the case type is of type series, the selection for the Input parameter is similar to the selection of the input parameter for the single point case input as described above. Note that upon using the Manual Case Control component to set series case input data a new tab appears where the values for the series can be created/set. The title of the tab sheet is consistent with the selected case mode selection. The graph below shows the (Steady State) series input for 2 input parameters. Enable the checkbox to create an automatic break in the output table.

A + − × Add point Renumber Graph

Point	Break	Manual Fuel Control Wf [kg/s]	Ambient Conditions Mach [-]
1	☑	0.380	0.800
2	☐	0.300	0.600
3	☐	0.220	0.400

Several `Options` are available to control the behavior of the input parameters:
`General Model Options`

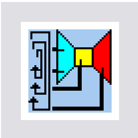
- `Auto reset off-design input to DP at DP case calculations`
  Reset the off-design input to design values for design point calculations
- `Set controlled values back to original after simulation run`
  When enabled, after the simulation, the input values that were in the numeric input fields before the simulation will be restored. The model will be in exactly the same state as before the simulation.

`Series Options`

- `Auto-config model St.St. series step options`
  Automatically adjust model Transient/Series Options options. More specific, this option

will cause the simulation to use integer values for `Point` numbering. The calculation series option will start at point 1 and have an increment of 1 to output data to the output tables. Note that this option will also renumber the series input data grid as it is required to have integer `Point` values. Intermediate points can and sometimes must be calculated for the sake of simulation stability. Extra intermediate points can be specified by the number of extra calculations in `Nr. of iteration intervals per point`. This will set the calculation step size (`Simulation step` in Transient/Series Options); increasing the number of intermediate calculations increase iteration stability to the next integer point. Unchecking this option will give the modeler an even higher flexibility to create the series by specifying custom input schedules. Note that the modeler then has to define the Transient/Series Options options manually!

8.4.3.3 Loop Case Control



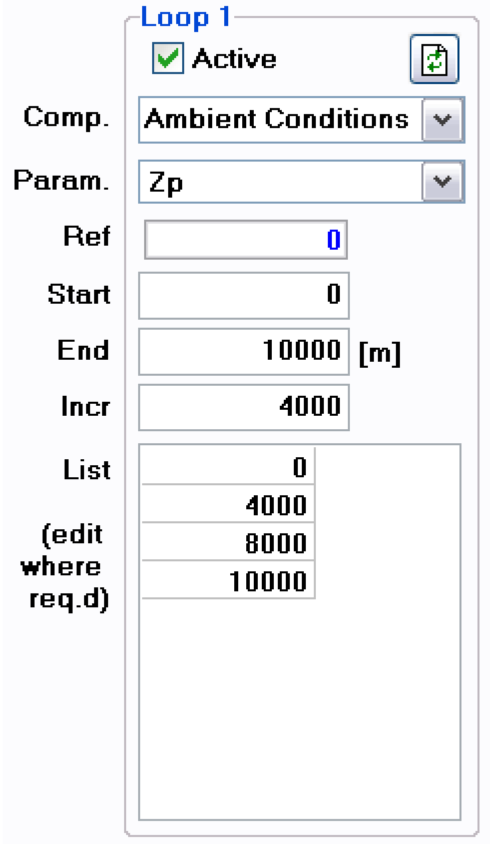
The Loop Case Control can be used as the central storage component for specific case input data. Effectively, the case input parameters in this component override the values set in the component data window. This component can create looped input series for a maximum of 3 input parameters.

In order to set the correct input for the model the modeler needs to set up the model so that the input that is required to be set by the Case Control Component is available in the component data window. If a parameter is not available it cannot be selected in the Case Control Component.

The figure below show a single loop case data entry interface. Similar to the Manual Case Control component the modeler sets up the component type and the input parameter when the `Active` check mark has been set. When selected, a reference value (`Ref`, non editable, bright blue font) will be displayed. Define `Start`, `End` and `Inc(rement)` values to automatically create the `List`. The numerical grid list can be edited using the following keys:

- Insert key - Insertion of a new row
- Delete key - Deletion of the active line
- Arrow keys - Navigate through the grid (note that if the active row is the last row, the "down arrow key" will add a new row to the grid)

To revert an edited list press the refresh button located at the top right of the input parameters group box.



When a maximum of 3 loop input parameters have been defined press the `Generate Series` button to create the looped input data sequence. The loop order is:

- Loop 1
  Outer loop parameter
- Loop 2
  Second or inner loop parameter
- Loop 3
  Inner or innermost loop parameter

Note that the sequence can also be edited like described above.

Several `Options` are available to control the behavior of the input parameters:
`General Model Options`

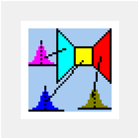
- `Auto reset off-design input to DP at DP case calculations`
  Reset the off-design input to design values for design point calculations
- `Set controlled values back to original after simulation run`
  When enabled, after the simulation, the input values that were in the numeric input fields before the simulation will be restored. The model will be in exactly the same state as before the simulation.

`Series Options`

- `Auto-break`
  When enabled, upon generating the sequence from the loop parameters, breaks will automatically inserted on every start of the innermost loop.
- `Auto-config model St.St. series step options`
  Automatically adjust model Transient/Series Options options. More specific, this option will cause the simulation to use integer values for `Point` numbering. The calculation series option will start at point 1 and have an increment of 1 to output data to the output tables. Note that this option will also renumber the series input data grid as it is required to have integer `Point` values. Intermediate points can and sometimes must be calculated for the sake of simulation stability. Extra intermediate points can be specified by the number of extra calculations in `Nr. of iteration intervals per point`. This

will set the calculation step size (`Simulation step` in Transient/Series Options); increasing the number of intermediate calculations increase iteration stability to the next integer point. Unchecking this option will give the modeler an even higher flexibility to create the series by specifying custom input schedules. Note that the modeler then has to define the Transient/Series Options options manually!

8.4.3.4 Monte Carlo Case Control



The Monte Carlo input controller is similar to Manual Case Control component. The difference is that a standard deviation can be specified to allow variation of the input value. Note that multiple input parameters can be specified. A random generator is used to create a user defined amount of input series using an inverse normal cumulative distribution function to calculate the input parameters based on the actual mean values and their respective standard deviations.

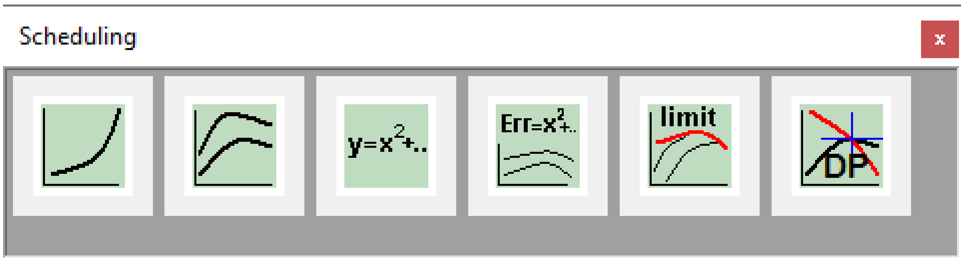
Note that the `Nr. of iteration intervals per point` can be used to speed up the calculations by choosing a small value. Although a decrease in calculation step size increases iteration stability, a relative large step can be chosen since the input variations are generally small for the solver.

8.4.4 Scheduling Component library

The Scheduling Component Library includes components for setting relations among parameters in order to force the engine to operate at specific operating points or within specific limit schedules. This is convenient for simulations where the control laws still need to be defined or are not available or when 'ideal control scheduling' is required. The following components are available :

- Equation/Schedule control components
  - 1-D Lookup Table Scheduler (DP via property and OD)
  - 2-D Map Scheduler (DP via property and OD)
  - Equation Scheduler (DP via property and OD)
  - Generic Schedule Control (DP via property and OD)
- Limiter (OD only)
- Design Point Equation Control (DP only)

The Schedulers can be used for DP and/or OD cases depending on their type. The Equation/Schedule control components and derivatives can be used for OD and DP, but for DP only if controlling a property (not an output variable).



8.4.4.1 Schedule Control components

There are four control components available to schedule a GSP output parameter or component property. During simulation with an active scheduler control component, GSP is adapting a user assigned free state variable (such as fuel flow) iteratively until the schedule parameter is at the specified value. The latter means an active schedule control component adds an error equation.

- Equation Scheduler for scheduling GSP output parameters or component property using an equation,
- 1-D Lookup Table Scheduler for scheduling a GSP output parameter using a 1-D table,
- 2-D Map Scheduler for scheduling GSP output parameter or component property using a 2-D map (specified in a map).
- Generic Schedule Controller, which combines the functionality of all 3 above components enabling the modeler to create an equation using an expression including parameters representing table lookup and/or map lookup output. Note that this component is recommended for advanced users only.

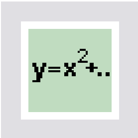
The parameter to control can be selected in the `Scheduled parameter` group box as:

- `Output parameter` (this option always requires to specify a state elsewhere) Scheduling output parameters is the straight forward way to schedule the engine model operating point. Select Output parameter (parameters specified as Output in component output tab sheets and Global output parameters) using the combo box.
- `Component property` (requires state depending on "Property is State" option) Scheduling GSP properties is for advanced usage and requires understanding of the GSP modeling internal details. GSP properties represent internal model parameters. The leftmost (active) combo box must be used to select the component from which to select the property, the right 2 combo boxes define what kind of property must be controlled:
  - `OD` Select component Off-Design property
  - `DP` Select component Design property corresponding to Off-Design property. If scheduling is also required for Design point calculation, then either the `Property is state` option or the `Determinate relation (no equation)`.
- The `Property is state` option is to avoid the need for an additional free state to match the number of states and errors. In this case the property itself will be the free state and then the solver will simply adjust this state variable (i.e. the property) to match the specified value. As such, the schedule equation added can be considered a 'dummy' equation simply saying the property (and corresponding state) must equal a given value.
- The `Determinate relation (no equation)` is for Property control only option and inhibits the adding of an error equation and state. This means during simulation the property is simple set by GSP and any effects from the changing property value only works on components calculated later that the Schedule control components itself. Although this option may well increase iteration speed and stability (reducing the number of equations), care must be taken with the calculation order of the components, since the property value will not affect components with higher calculation order numbers.

If the Project options | Advanced | `Show advanced model equation controls options in components` option is checked two additional options are visible in the General tab sheet:

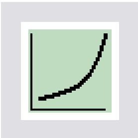
- The checkbox `Always create error equation` defines whether during Design point calculation even if not active (Active checkbox) an equation is added to the equation system.
- With the `Corresponding state nr. for deactivation` number the state with that number will be deactivated. Run the States and Error report to show current model states and errors (option can be found in the `model data` menu item of the models main menu).

8.4.4.1.1 Equation Scheduler



The *Equation Schedule Control* component allows the specification of an analytical expression using any output parameter to schedule either an output parameter or component property (i.e. the Scheduled parameter).
Create the analytical expression with the equation parser and the drop down parameter list.

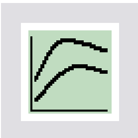
8.4.4.1.2 1-D Lookup Table Scheduler



The *1-D Table Look-up Schedule Control* component allows the specification of a simple 1-dimensional table (in the Table tab sheet) to schedule the selected output parameter or component property as a function of any Table input parameter. Table interpolation during simulation may be either linear or quadratic.

Select the table input parameter from the dropdown list on the `Table` tab sheet. The value for the output parameter or component property will be interpolated from the table (or extrapolated, depending on the interpolation option).

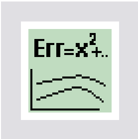
8.4.4.1.3 2-D Map Scheduler



The *2-D Map Schedule Control* component allows the specification of a 2-dimensional map (in the Map tab sheet) to schedule the selected output parameter or component property. The map is read from a file (GasTurb/MTU map format similar to the compressor maps etc.). Both an X and Y map input parameter need to be specified also: these are selected from lists with Output parameters. Map interpolation during simulation may be either linear or quadratic.

Select input parameters X and Y from the drop down lists on the `Map` tab sheet. The value for the output parameter or component property will be inter/extrapolated from the map.

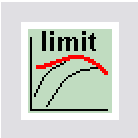
8.4.4.1.4 Generic Schedule Control



The *Generic Schedule Control* combines the functionality's of the *Equation*, *1-D Lookup Table* and *2-D Map Schedule Control* components into a single component for more complex schedules and equations among parameters. The expression in the General tab sheet represents the primary schedule but now may include one or two extra parameters: 'TABLEOUT' and 'MAPOUT' representing the output of the 1-D table and 2-D map schedules respectively.

The table and/or map data must be properly specified if the TABLEOUT and/or MAPOUT aliases are used in the expression.

8.4.4.2 Limiter



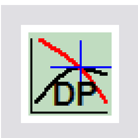
The Limiter control component limits a user specified GSP output parameter to a predefined limit schedule during an OD simulation. A power setting component must be specified (specified on the `Design` tab sheet) in order to inform GSP what parameter to adapt in order to 'stay within limits'.
Multiple limiters can be used working on different parameters. Complex limit schedules can be specified by combining an expression with the 1-D lookup table and/or the 2-D map, similar to the schedule components.

The limiter always works using an extra equation that may be either active or inactive. At the start of every OD steady state or transient operating point calculation, the limiter equation is inactive. After convergence, the limit parameter is compared to the limit schedule. If exceeding, the operating point calculation is restarted with the limiter equation active (the equation error then is the deviation from the limit schedule) and the input of the user specified power setting component (in the Design tab sheet) becomes a state variable. The resulting operating point after the 2nd calculation will be exactly on the limit. Note that limiters, when active, require additional (roughly double) execution time.

Note that when active, any other scheduling (for example during steady-state series and transient calculations by the transient input tables) of the power setting is overruled by the Limiter component.

Hint: the Limiter does not work in Design point mode. It is suggested to use the Design Point Equation component for scheduling to limiting relations in Design Point mode.

8.4.4.3 Design Point Equation Control



The Design Point Equation control is dedicated to perform parameter sweeps in Design Point (DP) simulations. It works similar to the off design Generic Schedule Equation control component. Complex relations among parameters can be specified using (parameter) expressions and can consist of 1-D and 2-D table look up values. For the free state property, an initial value needs to be given to the solver by the modeler (minimum and maximum values are optional). The solver will change this initial value as such that the expression is satisfied. Note that the numeric input data fields of the free state property are always specified in SI unity. Multiple DP equation controllers can be added to a configuration and each will add an equation to the solver system.

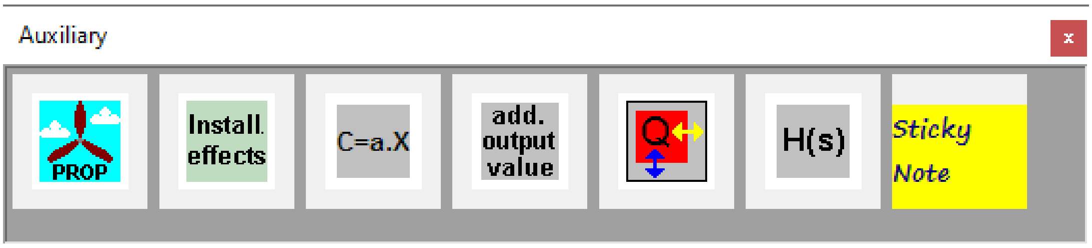
The DP equation control components can be used in conjunction with Case control components to specify Design Point series loops for the DP controlled parameter. Then the case controller controls the Design Point Equation control value.

This control may also be used if operation along limits is to be simulated since the Limiter component only works OD.

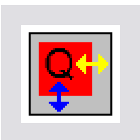
8.4.5 Auxiliary Component library

The Auxiliary Component Library includes the following components:

- Heat Sink
- Propeller
- Installation Effects
- Constant Expressions
- Additional Output Parameter



8.4.5.1 Heat Sink



Use the Heat Sink component to model heat transfer among gas path components or with the ambient environment. Add a heat sink component to the model and connect gas path components to the heat sink in the Heat Sink tab sheet of the gas path component. In the gas path component heat sink table, specify the specific heat transfer model data, one connection per row.

`General` tab sheet

- `Enabled`
  Enable/disable the `Heat Sink`. Disabling means all heat transfer with this heat sink is inhibited.
- `Dynamics`
  Define the `Effective mass` and `Specific Heat` of the heat sink. Note that these data only affect transient performance, similar to the heat soakage effect model.
- `Iteration control`
  Define a `Temperature 1st guess` and a `Q total error normalization factor` to optimize iteration performance and stability. Since the heat sink temperature is a state variable, For the

temperature, take the expected material temperature of the structures between the components, for example the gas path temperature averaged over the connected components. For the `Q total factor`, take a value in the order of the expected heat flux in order to have well conditioned Jacobian matrix for the solver; this is because the error equation for the heat sink is the sum of all heat fluxes being equal to zero

`External heat transfer` tab sheet
Here the heat transfer model data to the ambient environment and to other heat sink components can be specified in two separate tables, one per table row. See information below on parameters.

When the Heat Sink option is activated in the `Heat Sink` tab of a gas path component, a heat flux to or from the attached Heat Sink component is calculated. The Heat Sink component itself must also be activated. It is possible for a component to have more than one Heat Sink connected to it. This can be seen as different heat transfers that occur with different components. Heat Sink components can also be interfaced with multiple gas turbine components as well as with other Heat Sink components and the ambient environment.

Using Heat Sink components, complex thermal network models can be built connecting multiple heat sinks and gas path components. For steady state calculations, the heat capacity (mass and specific heat) does not affect the results since the system is in a thermal equilibrium. For transient calculations, the dynamic effects of heat transfer (heat soakage) is determined by the effective mass and the specific heat assigned to the Heat Sink component. Note that this heat soakage effect is in addition to the heat soakage effect that can be applied in the gas path component itself.

The heat sink model parameters are specified at two locations:
- Heat sink tab sheet of the connected Gas path component. Here the heat transfer model data for the heat transfer between the specific component and the heat sink are given in the table in the Heat sink tab sheet.
- Heat Sink component itself: here the general heat sink properties (mass, specific heat) and data for heat transfer data with the ambient environment and other heat sinks are specified.

An overview of the generic equations used is given in Heat sink equations. For a glossary on parameter names, also used in the sections below, see Heat sink parameters.

External heat transfer
In the `External heat transfer` tab sheet the data for modeling heat transfer to ambient can be specified. In the table rows multiple heat transfer models can be specified. Total heat transfer is the sum of all rows. Per row, either Ambient conditions as specified in the global model Ambient/Flight conditions can be selected or custom conditions can be selected in which case Temperature, Density, Mu and Cp must be user specified in the adjacent columns. A shaft suffix, D, Mu and density can be specified to calculate a Re number for rotating disks or cylinders. This option can be used to represent heat transfer with rotating elements in the gas turbine. Without a shaft suffix, Re will be 0 and the Re D or Nu L parameter now will represent the characteristic length L for the Nusselt relation. Next parameters to calculate Prantl number Pr (k gas) and the Nusselt expression for convective heat transfer is specified. This expression may include the 'Re' and 'Pr' parameters calculated before. Avoid Re being 0 (no shaft) if Re is used in the expression.
From the Nusselt expression Nu number is calculated. The convection heat transfer is then calculated using h_conv = Nu * k_gas / L. The conduction heat transfer coefficient is calculated with h_cond = k mat / d mat. Total heat transfer coefficient for convection and conduction then is h total = 1 / (1/h_conv + 1/h_cond). Corresponding heat flux Qconvcond = h total * Aht * (Tambient - Theatsink). Note that Qconvcond is negative when heat flows from the heatsink to ambient.
A wall surface temperature is calculated using Twall = Theatsink + Qconvcond / Aht / h_cond.

With an emissivity index Eps rad > 0, a radiation heat flux is calculated using
Qrad = (Twall^4 - Tambient^4) * Cstefanboltzmann * Eps rad * Aht.
Finally, total heat flux Qtotal then is the sum of Qconvcond and Qrad.

User specified heat flux

Heat transfer with the Heat Sink component also allows for a user specified heat flux. If the option is enabled a value can be specified in the field of the total heat rate "Qtotal" column at the far right of the table (check the User spec Q column) and calculations to determine the heat transfer coefficient are performed only for reference is possible and results are overridden by the user specified value. The GSP solver will iterate towards an operating point including heat sink and wall temperatures where this condition can be met. For example, if heat transfer of the heat sink with the ambient environment is user specified, then the heat sink temperature will go to a level where this heat flux can be realized by convection and/or radiation. Consequently, the heat transfer between gas path components and the heat sink will be affected.

Heat transfer among Heat Sink components

Heat Sink components can also exchange thermal energy among each other. This is an option that resides in the Heat Sink component. If heat transfer between two Heat Sink components is enable in a Heat Sink component, this is automatically enabled in the target Heat Sink. A user defined heat transfer coefficient with the unit [W/K] can be specified for a heat flux proportional with the temperature difference between the heat sinks. This corresponds to heat transfer by conduction between solids. Therefore, the heat transfer coefficient can be estimated by:

$$heat\ transfer\ coefficient \approx \frac{kA}{\delta}$$

Only in one of the 2 heat sink components a value for the heat transfer coefficient must be defined. The other heat sink automatically accepts this heat transfer and does not require additional input.

After calculation, the total of the External conditions heat transfers heat fluxes and the Heat fluxes to other heat sinks is summed up to provide the total heat flux to/from the component. During steady state this sum must be 0, during transient, it can be non-zero only if a heat sink mass and Cp has been specified in the `Dynamics` box in the `General` tab sheet. This heat balance is added to the GSP equation system while the heat sink temperature is added as a state variable.

8.4.5.1.1 Heat sink parameters

In the following two lists of parameters that pertain to the Heat Sink options, either for a component Heat Sink tab or for the Heat Sink itself, the bold face printed variables are calculated, whereas the rest of the variables are user defined. For the Heat Sink component there is one exception. When the external conditions are set to the ambient/flight conditions the values for the temperature, density, kinematic viscosity, and the specific heat are calculated from the Flight / Ambient conditions.
While most parameter units and display formats are controlled by the unit system and Output format options settings, this does apply to the following: k (conductivity), h (per area unit heat transfer coefficient) and H (heat transfer coefficient). These parameters have fixed units and formats, indicated by the 'fixed unit' in the lists below.

Parameters in the Heat Sink tab in a gas turbine component
Multiple heat sinks can be specified in multiple rows of the **Heat sinks table**. The resulting total heat flux to/from the component is the sum. of the heat fluxes of all rows. Table cells are editable only if they represent user specified model input and are not overridden by User spec Q option. If readonly, the cells have a blue font color.
Heat Sink Component: The heat sink component that is connected to the current component.

A_ht	The area for heat transfer with the Heat Sink component.
A_flow	The area of the gas path. It is necessary for the calculation of the flow velocity and must therefore be the total area available for the total mass flow.
D_re	The (hydraulic) diameter; needed for the calculation of the Reynolds number. In case of a compressor this should be the area of one single cascade passage otherwise turbulence will not be well represented.
Re	The calculated Reynolds number.
k_gas	Thermal conductivity of the gas, fixed unit [W/mK].
Nu_eq	The Nusselt number relation for the specific type of convective heat transfer.
Nu	The calculated Nusselt number.
d_mat	The characteristic/average distance from the center of the wall material to the gas path (the average heat travel length through the material).
k_mat	The thermal conductivity of the material of the current component connected to a Heat Sink component, fixed unit [W/mK].
h_total	The calculated total heat transfer coefficient, fixed unit [W/m2K].
Q_convcond	The calculated heat rate obtained by convection and conduction combined.
Eps_rad	The value for the emissivity of the Heat Sink component [-].
T_wall	The calculated wall temperature for radiation calculations.
Q_rad	The calculated heat rate due to radiation.
Q_total	The summation of Qrad and Qconvcond for the Heat Sink component. A positive value indicates heat transfer into the component, whereas a negative value indicates heat transfer out of the component.
User spec Q	This is an option that enables user defined heat rates for the heat sink.

Parameters of the Heat Sink components

Multiple connections to external/ambient conditions can be specified in multiple rows of the **External conditions heat transfer table** and the resulting total external heat flux to/from the heat sink component is the sum of the heat fluxes of all rows. Table cells are editable only if they represent user specified model input and are not overridden by User spec Q option or External conditions option. If readonly, the cells have a blue font color.

External conditions: These are the external conditions to which the Heat Sink component transfers heat when this option is enabled:

Shaft suffix	This option links the Heat Sink to a specific shaft in order for it to have a rotational rate. Typically used when the Heat Sink component represents a rotating Disk.
T_ext	The temperature for which the total heat rate to the exterior is calculated, either obtained from the Flight / Ambient conditions or user specified depending on the External conditions option selected.
A_ht	The area for heat transfer with ambient.
D_re / L_nu	The diameter of the Heat Sink component and/or characteristic length L for the Nusselt number [m]. It is necessary for the calculation of the flow velocity of the airflow over the Heat Sink and calculating the Reynolds number Re for convective heat transfer in case of rotation and/or for calculating the heat transfer coefficient from the Nusselt number.
Rho_re	The density of the gas flow over the Heat Sink component,
Mu_re	The kinematic viscosity necessary for the calculation of the Reynolds number.
Re	The calculated Reynolds number.
k_gas	Thermal conductivity of the gas, fixed unit [W/mK].
Cp_gas	Specific Heat of the gas used for the calculation of the Prandtl number.
Pr	The Prandtl number.
Nu_expression	The Nusselt number relation for the specific type of convective heat transfer.
Nu	The calculated Nusselt number.

d_mat	The characteristic/average distance from the center of the Heat Sink to the exterior (the average heat travel length through the material).
k_mat	The thermal conductivity of the material of the Heat Sink component, fixed unit [W/m K].
h_total	The calculated total heat transfer coefficient, fixed unit [W/m2K].
Q_convcond	The calculated heat rate obtained by convection and conduction combined.
Eps_rad	The value for the emissivity of the Heat Sink component, [-].
T_wall	The calculated wall temperature for radiation calculations.
Q_rad	The calculated heat rate due to radiation.
Q_total	The summation of Qrad and Qconvcond for the Heat Sink component. A positive value indicates heat transfer into the component, whereas a negative value indicates heat transfer out of the component.
User spec Q	This is an option that enables user defined heat flux. When checked, the value entered in the Qtotal column will be used and not overridden by calculation from the other column data.

Multiple connections to other heat sink components can be specified in multiple rows of the **Other heat sinks table** and the resulting total external heat flux to/from the heat sink component is the sum of the heat fluxes of all rows.

Other Heat sink	: Select the other heat sink to which this heat sink is connected.
H	Heat transfer coefficient H in W/K for calculation Q using Q = H * (Theatsink - Totherheatsink); fixed unit [W/K].
Q	The heat flux to the other Heat Sink component. A positive value indicates heat transfer into the component, whereas a negative value indicates heat transfer out of the component.
User spec Q	This is an option that enables user defined heat flux. When checked, the value entered in the Qtotal column will be used and not overridden by calculation from the other column data.

8.4.5.1.2 Heat sink equations

Heat transfer by convection is characterized by the Reynolds and Prandtl numbers Re and Pr:

$$Re = \frac{\rho u D}{\mu} \qquad Pr = \frac{\mu C_p}{k}$$

‘D’ is the variable ‘Re D’ given in the Heat Sink component variables list. Unless stated otherwise, this remains the case for subsequent equations with the variable ‘D’. For channels with a non-circular cross-section, ‘D’ is the hydraulic diameter. For the Reynolds number the velocity u is not defined and must therefore be calculated. The flow velocity depends on the flow type. For the internal heat flux from gas path to heat sink (specified in the gas path component) the mean flow velocity is determined using:

$$u = \frac{m}{\rho A_{channel}}$$ which is the continuity equation.

In the special case where shaft suffix is specified in the heat sink external heat transfer to ambient table, convection is assumed to be enhanced by rotation. Then the velocity of the external wall is

$$u = \frac{\Omega_{shaft} 2\pi}{60} \frac{D}{2}$$ and Re and Pr can be calculated. In this relation *omega* is the shaft angular velocity of the shaft to which Heat Sink is connected. ‘D’ is the 'rotating diameter' of

the Heat Sink component.
Note that for the normal case of a static outside wall, natural convection is assumed and Re is not active unless a constant Re is specified to represent forced convection. Then the user must specify a Nusselt relation without Re in the expression.

For heat transfer to the surrounding environment the Prandtl number is determined from the user defined thermal conductivity and kinematic viscosity of the gas. The specific heat Cp is calculated from the user specified temperature of the environment. If the external conditions are set to the ambient/flight conditions the variables required for the Prandtl number are calculated by GSP.
The heat transfer coefficient used in the calculations is not the regular heat transfer coefficient, but rather a combination of the convective heat transfer coefficient and the conductive heat transfer coefficient. This relation is given by

$$\boldsymbol{h}_{total} = \frac{1}{\frac{1}{\boldsymbol{h}_{conv}} + \frac{1}{\boldsymbol{h}_{cond}}}$$

The convective heat transfer coefficient is a function of the Nusselt number and the thermal conductivity of the gas,

$$\boldsymbol{h}_{conv} = f\left(Nu, k_{gas}\right) \boldsymbol{h}_{conv} = f\left(Nu, k_{gas}\right),$$

and the conductive heat transfer coefficient is a function of the thermal conductivity and the thickness of the solid material

$$\boldsymbol{h}_{cond} = f(k_s, \delta_s) \boldsymbol{h}_{cond} = f(k_s, \delta_s).$$

The convective heat transfer coefficient is determined with the Nusselt number.

$$\boldsymbol{h}_{conv} = \frac{Nu \cdot k}{D}$$

The user specified Nusselt number correlation is specific to the type of flow that occurs.

The heat transfer between the components, Heat Sink and surrounding environment is defined by heat transfer coefficients, interface areas and the temperature difference between the different components and/or the surrounding environment. The heat transfer coefficients are determined from the Nusselt number correlation defined and the parameters to determine the value of the Nusselt number. In this section the mathematical relations required to calculate the total heat rate “Qtotal”, are explained briefly.
In general heat transfers due to conduction, convection and radiation are calculated with the equations

$$Q_{cond\,(hot \to cold)} = \frac{k \cdot A}{\delta} (T_{\boldsymbol{hot}} - T_{cold}),$$

$$Q_{conv\,(fluid \to wall)} = \boldsymbol{h}_{conv} \cdot A \left(T_{fluid} - T_{wall}\right)$$

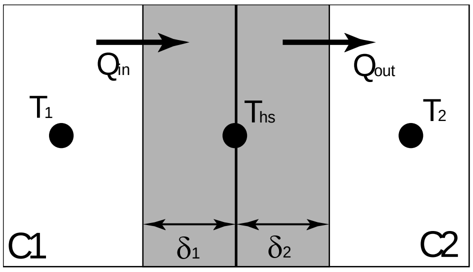
$$Q_{rad\,(wall \to \infty)} = \varepsilon \cdot A(T_{wall}^{4} - T_{\infty}^{4})$$

respectively. The positive directions of heat transfer are indicated by the subscripts of the heat rate Q. The temperatures in these equations are the hot and cold body temperatures of two solids, in case of conduction, or the mean fluid and wall temperature, in case of convection, or the temperature of the wall and surrounding environment in the case of thermal radiation.

Conduction and convection
In GSP the temperatures are defined slightly different. If two different components are connected to each other via a Heat Sink component and operate under steady state conditions, the heat flow Q must be a value such that no heat is stored within the system, i.e. Qin = Qout or in equation form

$$A_1 h_1 (T_1 - T_{hs}) = A_2 h_2 (T_{hs} - T_2)$$

A1 is the interface area between component C1 with temperature T1 and the Heat Sink, h1 is the total heat transfer coefficient defined above. The same holds for component C2.



Graphical representation of one Heat Sink connected to two component: C1 and C2.

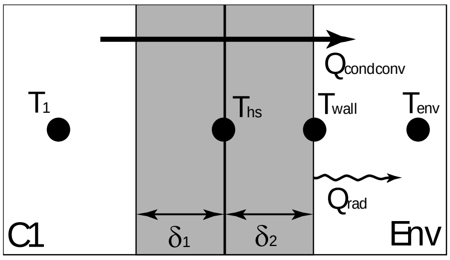
The figure above shows a graphical representation of the connection of two components and one Heat Sink. The temperature difference between the component C1 and Heat Sink component is given by (T1-Ths). In a regular heat transfer problem consisting of a convection-conduction-convection mechanism, the temperatures of the gas flows could be known and the wall temperatures and heat transfer rate of the conductive material have to be determined. In the GSP model the solid material has a single temperature, Ths. This temperature is determined such that Qin = Qout, and can be visualized as an average or bulk temperature of the two wall temperatures.
The conductive layer thickness, delta_i, is defined such that it represents the distance from a connecting component to the center of the Heat Sink. To have a good representation of actual heat transfer, the values for the conductive layer thickness of components having a mutual heat transfer must be such that the sum matches the actual thickness.

Radiation
The above analysis includes only conduction and convection. If radiation is also included in the simulation, the wall temperature needs to be calculated using

$$Q_{condconv} = h_{conv} A (T_{env} - T_{wall})$$

which can be rewritten to $$T_{wall} = T_{env} - \frac{Q_{condconv}}{h_{conv} A}$$.

The overall heat transfer model including radiation can be represented by the following figure



Graphical representation of thermal radiation of the environment in combination with convective and conductive heat transfer

Introducing thermal radiation into this model changes the thermal equilibrium of the heat sink. The correct Twall is obtained during the iteration towards the gas turbine operating point. The emissivity epsilon of the material is a dimensionless quantity which has a value between 0 and 1. It is the ratio of energy actually radiated by a particular material to the energy radiated by a black body (a hypothetical object that absorbs all electromagnetic radiation that falls on it) at the same temperature.

8.4.5.2 Propeller

The propeller model component can not only be used to model both fixed pitch as constant speed propeller applications using a specific propeller map format, but also can be used if no propeller map characteristics are available.

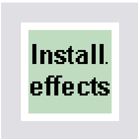
Propeller type can be specified

- `Prescribe efficiency`
  The 'Prescribe efficiency' model uses the ideal (or actuator disc) propeller model to calculate maximum total efficiency (i.e. propulsive efficiency) at which any user specified value exceeding it is truncated. For conceptual design studies where the propeller design has not yet been defined fully (and no map is available), using the ideal propeller model is often the best option. For the ideal propeller theory, refer to the technical (TM) GSP manual. Specify a propeller efficiency to represent a reasonable loss, both for in-lfight and static (Vt=0, and so advance ratio J=0) conditions.
- `Map / fixed pitch`
  With a fixed pitch propeller with known D and pitch angle, use a propeller map and manually specify the propeller pitch (beta) angle and also the design point in the map. For map scaling, often it is best to specify DP at in-flight conditions. Note that setting up a model where the power turbine is configured **with** a power balance equation, power setting Wf as free state and power turbine / propeller speed is user specified is the preferred model setup. In this model setup, the model finds the required power level to maintain propeller speed. In fact, now fixed propeller speed is the power setting input variable in this configuration. Although alternative configurations are possible (i.e. propeller speed as free state instead of Wf) these usually are **much less stable** and **not recommended**.
- `Map / pitch control`
  With a variable pitch ('constant speed') propeller, use a propeller map and use a propeller control to specify propeller pitch (beta) angle. Make the propeller control input a free state and configure the power turbine with the power balance equation. Power setting input is Wf or TIT (TT4). The propeller pitch angle will then be found corresponding propeller torque and power matching the power turbine delivered power. Set OD propeller speed by adapting

power turbine speed. For map scaling, often it is best to specify DP at in-flight conditions, such as 1000 m / Ma 0.2 in this example model configuration.

Warning:
Note that the `Prescribe efficiency` propeller model option has been introduced and added on top of the existing propeller options in version 11.4.4.0! One may need to update propeller component data from older/existing models.

8.4.5.3 Installation Effects



When installing the engine in an aircraft the thrust of the engine as a system reduces due to several drag effects. These drag effects are calculated post-convergence and added to the net thrust (FN) to obtain the installed net thrust (FNinst). Use the global output options to select the installed thrust output parameter.

Current modeled effects include:

- Spill Drag
  The spill drag calculation is based on the following equation: Dspill = CdSpill * Highlight Area * Dynamic Pressure,
  wherein CdSpill is the spill drag coefficient which is a function of flight Mach number and area ratio (area ratio is defined as free stream area/intake highlight area, which usually is A0/A1) and the dynamic pressure is defined as ½ * Rho * V^2. The spill drag coefficient will be read from a separate map for values of Mach and area ratio.

  The checkbox enables/disables the calculation of spill drag.

  The highlight area can be specified as:
  - `Area`
    user defined value
  - `Total inlet area`
    total calculated inlet area (could be from more than 1 inlet component)
  - `Outp. param.`
    select an output parameter (e.g. a calculated column or an Additional Output Parameter)

  The free stream capture area can be specified as:
  - `A0 (total free stream area)`
    calculated value based on number of inlets, and the equation of continuity
  - `Outp. param.`
    select an output parameter (e.g. a calculated column or an Additional Output Parameter)

  Use the map functionality to select a spill drag coefficients map.

- Afterbody Drag
  The exhaust nozzle drag (afterbody drag) calculation is based on the following equation: Daftbody = CdAftbody * Reference Area * Dynamic Pressure,
  wherein CdAftbody is the afterbody drag coefficient which is a function of flight mach number and Nozzle Pressure Ratio (NPR) and the dynamic pressure is defined as ½ * Rho * V^2. The afterbody drag coefficient will be calculated by reading the CdAftbody from a separate map for values of Mach and Nozzle Pressure Ratio.

  The checkbox enables/disables the calculation of afterbody drag.

The reference area can be specified as:

- `Area`
  user defined value
- `Aexit (total exhaust area)`
  total calculated exhaust area (could be from more than 1 exhaust component)
- `Outp. param.`
  select an output parameter (e.g. a calculated column)

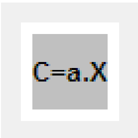
The NPR can be specified as:

- `Exhaust comp.`
  NPR is taken from the specified component
- `Outp. param.`
  select an output parameter (e.g. a calculated column)

Use the map functionality to select an afterbody drag coefficients map.

Several output options have been added to monitor the installation effects, which can be found on the `Output` tab sheet. Output options include drag values, drag coefficients, etc.

8.4.5.4 Constant Expressions

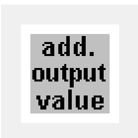


Constant expressions provide the user with a means to represent constant values with parameter identifier names. The modeler can define constants or constant expressions for an identifier that in turn can be used in any component model input field expression that is parsed during simulation. As an example, one could define a 'Scale' constant parameter that is multiplied with other values in expressions such as the inlet mass flow to represent different engine scales. With changing the scale in the constant expression component, the actual inlet mass flow will be scaled accordingly, which is convenient for scale effect studies.

It is advised to use a single constant expressions component per configuration/case and add multiple constants in the table. Constant expressions may use constants defined in prior table rows.

Note that it is easy to configure off-design and design modifiers and beware that changing a constant design expression requires a reset (new design calculation) of the model. Also beware that this will change the model (gas turbine) configuration.

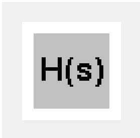
8.4.5.5 Additional Output Parameter



This component can be used to create an additional output parameter similar to a calculated column expression. Use the expression fields to define a custom parameter. The calculation itself, or the parsing of the expression, does not affect the engine model state and therefore will not not affect the solution and convergence. However, the parameter will process the expression during convergence and can thus be used to control other input of the model so that the parameter can affect the model state.

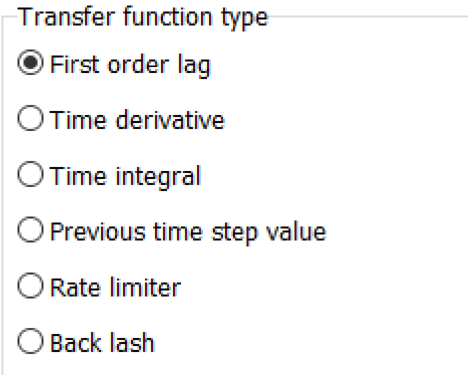
Note that the expression interface inherits the functionality from the Schedule Control components. This implies that the expression can consist the 'TABLEOUT' and 'MAPOUT' output parameters (available in the parameter drop down list) to use the table schedule and the map from the '1D-Table' and the '2D-Map' tab sheets (see also Generic Schedule Control).

8.4.5.6 Transfer_function

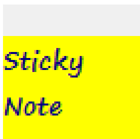


Transfer functions are to be used to represent dynamics of output parameter signals during transients. Especially useful to simulate sensor dynamics with parameters that are used in schedulers, limiters and equations.

Various transfer functions can be selected to alter the dynamic response:

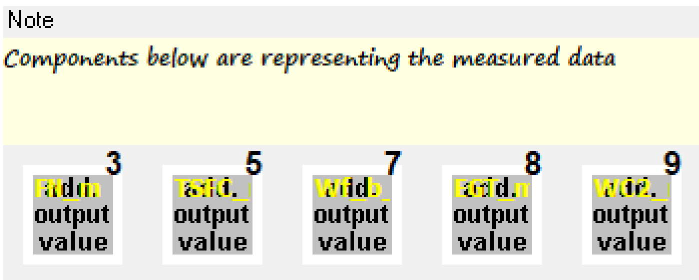


8.4.5.7 Sticky Note



The sticky note component allows for quick messages on the model form, these can be reminders for other modelers to pick up modeling where you stopped for instance.

E.g.:



8.4.6 Miscellaneous Components

The control components are arranged in libraries. The following libraries contain control components:

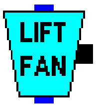
- STOVL Component Library
  Contains specific components for STOVL (Short Take Off and Vertical Landing) propulsion system modeling including lift fans and associated drive system components.

8.4.6.1 STOVL Component Library

The GSP STOVL Component Library is currently not maintained and therefore not included in the application.

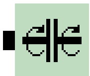
The STOVL Library includes the following 6 gas turbine components:
- Lift Fan
- Clutch
- Lift Fan Exhaust
- STOVL Convergent Exhaust Nozzle Control
- Lift Fan Inlet
- STOVL FADEC

8.4.6.1.1 Lift Fan



The lift fan component is used to represent the lift fan of a (turbofan) engine used in STOVL aircraft. The compression process is modelled entirely equal to the compressor component. A single fan map is used for off-design calculations. The lift fan has no compressor bleed options.

8.4.6.1.2 Clutch



The clutch component is used to transmit the power from the main drive shaft to the lift fan drive shaft in STOVL operation. Engaging and disengaging the lift-fan driveshaft from the fan shaft while the engine is running can be modelled.

A clutch is an unusual element in a gas turbine drive train system. However, there are a number of applications, where the load needs to be coupled and uncoupled from a gas turbine drive shaft during operation. In these cases, a clutch is required to allow smooth transfer from the uncoupled state to the fully coupled state and vice versa, without excessive torque loads on the shafts due to high acceleration rates.

Examples of clutches in gas turbine drive trains may be found in: vehicular turboshaft engines, helicopter drive trains and STOVL propulsion systems using liftfans driven by the main engine LP shaft.

In general, a clutch is a device that is able to transfer a certain (limited) amount of torque between two shafts. Several systems exist to transfer the torque including wet and dry friction plate systems and a variety of hydraulic systems (?).

For analysis of system performance, the clutch model minimally needs to accurately represent the torque transferred. If also clutch performance itself requires scrutiny, more detailed models may well be required. This section describes the clutch model as implemented in GSP (NLR Gas turbine Simulation Program, ref. xx), with calculation of torque transmission, clutch state and friction heat production.

For the clutch model, a number of terms/parameters are introduced required to determine the state of operation of the clutch.

1. *Engagement status*
   A clutch can be fully engaged of disengaged. If fully engaged, it is able to transfer maximum torque capacity; if fully disengaged, usually no torque is transferred unless some sort of residual friction loss is defined in the model.
   In GSP an 'engagement variable' is used, ranging from 0 to 1. 0 is fully disengaged, 1 means fully engaged.
2. *Locked/unlocked status*
   If a clutch is locked, both shafts run at the same rotor speeds and the clutch functions as a coupling. In this case, the torque transferred does not exceed maximum static torque capacity. If the clutch is unlocked, both shafts are not running equal speeds. There is a case where the clutch in unlocked at equal speeds, but this only can occur during a very short time of transition between the locked and unlocked states, or when at least one shaft is accelerating past the other shaft speed.
3. *Static torque capacity*
   Static torque capacity is the maximum torque the clutch can transmit in the locked state. This means the friction materials do not move (relative to each other) and the static friction coefficient applies. Static torque capacity always is equal or larger than dynamic torque capacity.
4. *Dynamic torque capacity*
   Dynamic torque capacity is the maximum torque the clutch can transmit when it is unlocked, i.e. rotor speeds are not equal and so the friction materials are moving relative to each other, so the dynamic friction coefficient applies. Dynamic torque capacity always is equal or smaller than static torque capacity.
5. *Torque demand*
   Torque demand is the torque that would be transmitted in the locked state. It can also be described as the torque that would exist in the shaft if maximum torque would be infinite and no slipping would occur.
6. *Slipping*
   The clutch is slipping if the two rotor speeds are unequal and engagement is larger than 0. Torque required exceeds maximum dynamic torque capacity and therefore cannot fully be transmitted. This means friction heat is produced proportional to the torque (engagement x maximum dynamic torque) and the delta in rotor speeds.

The operation mode of interest with a clutch model is transient. For steady-state simulation, either the fully engaged or disengaged state must be assumed. For system modeling environments like GSP, this means that the design point state also either is fully engaged or disengaged. Prior to a transient simulation, then a fully engaged or disengaged state must exist.

GSP implementation

In the GSP clutch component, a static and dynamic maximum torque can be specified. The engagement value, ranging from 0 at disengagement to 1 at full engagement, determines the actual torque capacity as a fraction of maximum torque.
The engagement value during a transient simulation can either be obtained from user-specified time functions or result from a control system model output. With a clutch control model, accurate simulation of clutch performance in complex systems including (closed loop) engagement can be performed.
Friction heat production due to clutch slipping is calculated, and with more data an accurate heat flow and conduct model can be added to analyze local heat loads and temperature levels during and after successive engagement events
In ref. GSP and example is given of simulation of a turbofan driving a STOVL lift-fan through a clutch. The lift-fan is driven by the main engine fan shaft through a dry Clutch that is able to disengage the Lift Fan from the engine during normal forward flight and engage during vertical flight modes. The GSP model and libraries required are included in the GSP registered version.

One of the challenges in modeling clutch engagement- or disengagement transients, is to determine whether static or dynamic maximum torque is applicable. In other words: is the clutch in a locked state or is it slipping. In the transient simulation this means the model algorithm needs to keep track of the clutch state (locked or not) history during transient simulation, i.e. save the clutch state of the prior time step. The state of the prior time step then determines whether dynamic or static maximum torque applies. For example, during a transient starting with a locked clutch but increasing torque demand, at some point this torque will exceed maximum static torque and the clutch will unlock and start slipping. After the time step where the unlocking has been determined, the lower level of dynamic maximum torque applies, and the clutch can only lock again after the torque demand will drop below dynamic maximum torque. If torque demand fluctuates around maximum static torque levels, severe transient load effects on the clutch can be simulated. In this case however, small time steps will be required to accurately represent rapid transient effects.

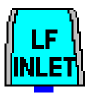
8.4.6.1.3 Lift Fan Exhaust

The lift fan exhaust can be used to model the exhaust of the lift fan.

8.4.6.1.4 STOVL Convergent Exhaust Nozzle Control

STOVL specific CENC which works together with the STOVL FADEC.

8.4.6.1.5 Lift Fan Inlet



The lift fan inlet component represents a STOVL inlet. The lift fan inlet is identical to the standard inlet component.

The entry conditions are taken from the ambient conditions window.

8.4.6.1.6 STOVL FADEC

STOVL specific FADEC which works together with the STOVL CENC .

8.4.7 Custom Components

NLR can provide customized libraries or components. These are separately developed program units (.bpl), that are dynamically loaded at GSP start-up when located in the same folder as the GSP.exe.

8.4.7.1 Custom Library

If detailed analysis of specific applications is required, NLR can provide dedicated custom components specific to your needs in a separate custom component library.

Custom library versions

Since custom libraries are also compiled separately, they have their own version number. Also see GSP versioning.
Usually, custom libraries will be provided and issued together with the standard GSP builds and included in a single installer and then the file version version number will be the same as the GSP file version number (GSP main window About box under help menu). However, custom libraries can also be separately installed or left in the GSP installation folder while a newer GSP build is installed and the custom library file not overwritten. In that case, file version numbers may be different and compatibility issues may emerge. So in case of problems with custom libraries, the custom library version number (status bar on the bottom) should be compared with the GSP file version number.

Please contact NLR regarding custom features in components.

8.4.7.2 Custom components

When detailed analysis into specific phenomena inside a gas turbine is required or for analyzing the effects of engine control logic (e.g. non-linear, multivariable control, specific Full Authority Digital Electronic Control FADEC), specially tailored custom component models are required. These may be new models based on existing component models adapted to specific needs or entirely new models for simulation of auxiliary systems such as generators, fuel systems, heat management systems etc.

Custom components are provided in separate custom libraries and require additional coding. Using the GSP Component Developers Package (CDP), custom components can easily be derived from the standard component models using object inheritance.

NLR has a large number of custom libraries available, developed for detailed performance analysis of specific engine designs and engine control systems. New custom libraries are usually developed at NLR. For advanced use of GSP, custom components can be developed outside NLR using the additional GSP Component developers package.

Contact NLR for additional information on Custom components or the GSP component developers package.

8.4.8 Component template libraries

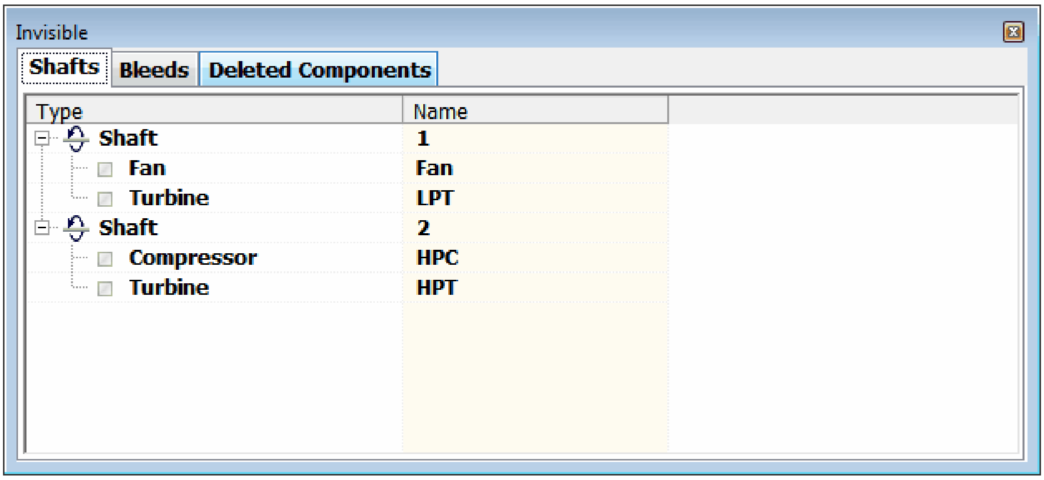
GSP users can create their own 'libraries' with pre-configured component models to be used as templates for new models saved to separate Projects. From this project's model panel, components can be copied into new or existing model panels of other projects.

8.5 Invisible components

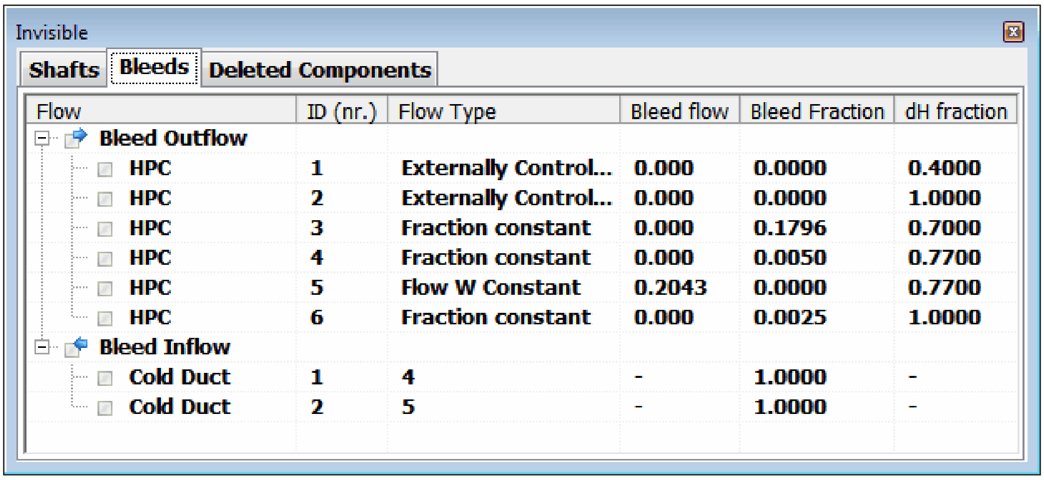
The major components of a gas turbine are represented by icons visible on the model window. However, some subsystem components or processes transmitting energy in the form of hot air or shaft power (shafts and bleed flows) are represented by model system objects. These objects are not visualized in the model window. The parameters defining the properties of invisible objects are always specified in a visible component model's data window.

An option `List Invisible Components` in the model menu is available to list the visible components. A window will be opened to display lists of shafts and bleeds. The window can be docked in the project window.

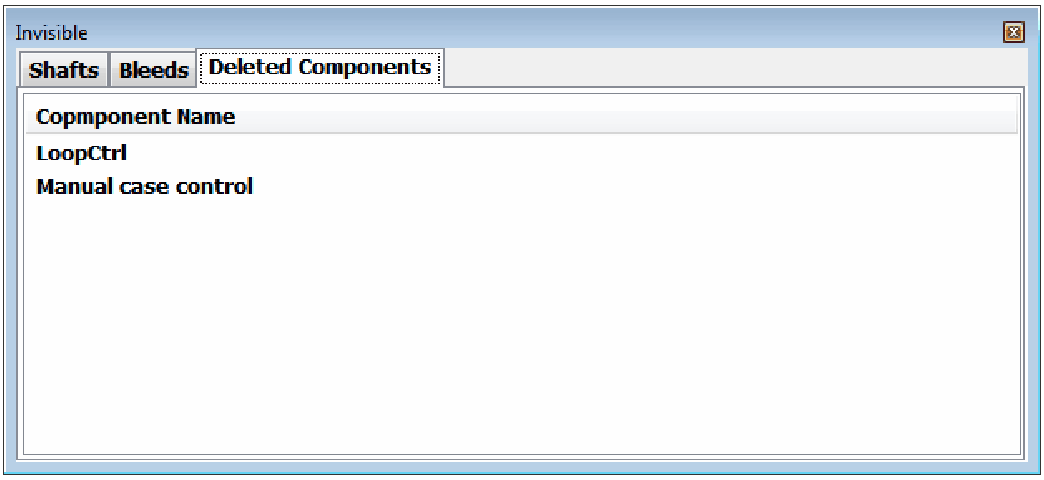
The `Shafts` tab sheet shows the hidden shaft components/objects and how these are connected.



The `Bleeds` tab sheet shows the hidden bleed in- and outflow ports and how these are connected.



The `Deleted Components` tab sheet displays a list of deleted case input components. Double-click on the entry to restore the component (only possible if there is no component of the same type present on the model window that have the "allow only one of this component type in the model" active, this is a developer design option, e.g. only a single case model component is allowed).



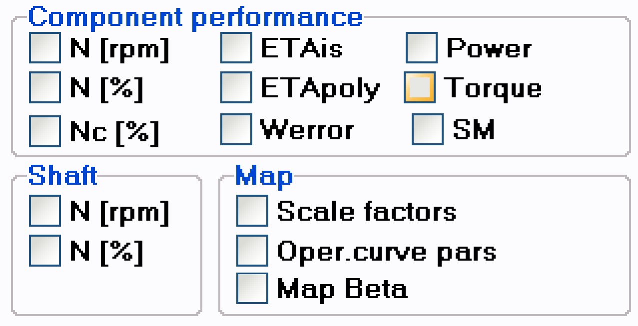
8.5.1 Shafts

Shafts transmit mechanical power between components such as turbines, compressors and pumps and are automatically defined in these components. To connect shafts, just specify equal shaft numbers or suffixes for two components in the data specification windows. For example, to connect a compressor to a turbine, set the `shaft nr.` equal in both `General` tab sheets. The turbine mechanical power will be added to the shaft while the compressor will absorb all available power from that shaft if the free state rotor speed option is set. Several options are available on how to handle "power surplus or deficit" in a shaft during simulation, depending on the component options set (see Turbine component).

Shafts are not visible on the model panel.

The relation between Shaft speeds and component speeds has changed, the first component defining shaft (usually fan compressor), defines shaft speeds. To output the shaft speed parameters separate Shaft (shaft nr. ID) and Component (component ID) rotor speed output parameters are available in turbo machinery component output tab sheets. Since duplicate output is not allowed, setting output data, for shaft speed for instance, should be done in one of the turbo components the shaft. Alternatively, use the central output functionality of the output parameter tab sheet by selecting component output for turbo components. Setting shaft properties for turbo components will automatically prevent duplicate output.

The figure below shows a snippet of the output parameters tab sheet of a turbo component (e.g. compressor, turbine, etc.)



Component parameters, such as rotor speeds and torques, can be selected in the top option box, which can be identified in the output table by the parameter name followed by an underscore and the component number (or the `ID string` from the component data entry window) (e.g. Nc_4 and PWshaft_4 are corrected speed and power input or output in component nr. 4), while shaft output can be selected in the lower left option box, which can be identified in the output table by the parameter name followed directly by the shaft ID *(e.g.* N2 means rotor speed in rpm of shaft nr. 2).

Shafts have two properties depending on engine station gas conditions: Nc and Tcorr. Shaft corrected speed Nc is defined as the shaft rotor speed corrected to ISA using the entry conditions of the FIRST turbomachinery component in the gas path on the shaft. This is important to note for components using the corrected shaft speed (and specifying shaft suffix) such as the Bleed schedule control component.

8.5.2 Secondary air and gas flows

Secondary gas and air flows include all non-main gas path flows such as compressor bleed flows, leakage flows, turbine cooling flows etc. Compressor bleed flows may be used as customer bleed (exiting the engine), anti-icing bleed (sometimes re-entering the engine) or turbine cooling bleed (re-entering the engine). Secondary air flow rates can be controlled by other components such as bleed control components.

Secondary air flows are identified with numbers (bleed flow numbers, turbine cooling flow numbers) and are not visible in the model window.

8.5.3 Bleed flows

Bleed flow invisible objects represent secondary air or gas flows among the components such as:

- compressor bleed used as customer bleed
- compressor bleed for turbine cooling
- compressor handling bleed (variable bleed)
- small leakage bleed flows flowing into the duct
- small leakage bleed flows flowing from the fan core
- duct bleed for customer bleed or cooling

In the Bleeds tab sheet in the compressor, duct or fan data window an unlimited number of bleed flows can be specified. The bleed flow numbers used must be unique for the model. Reusing numbers will raise a warning on simulation and stop calculation. The type of bleed flow can be selected from a drop-down list box activated when clicking in the grid cell on the arrow on the right. The following types are available:

- `None`
  The row in the table represents no bleed

- `Flow W constant`
  A constant flow rate in [kg/s] must be specified

- `Fraction constant`
  A constant fraction (in the 0-1 range) of the compressor inlet flow must be specified

- `Externally controlled`
  The bleed flow rate is controlled by an external component; e.g. a bleed control component

- `Determined by map` *(not in standard compressor, only available in derived compressor classes)*
  A bleed flow determined by the compressor operating point data in the map. The third map output table represents compressor inlet corrected bleed flow values. In effect, the bleed flow can be set depending on corrected compressor speed and beta value. It is a means to include variable handling bleed without using control system components. Handling bleed is usually controlled as a function of corrected compressor rotor speed only.

Note that the Bleed Nr and Type columns are only accessible in Configurations since they are considered configuration properties. The other columns can be adapted both in configurations and cased, both DP and OD.
For separate control of DP and OD bleed flows, use the `Externally controlled` type in conjunction with a bleed control component which allows separate DP and OD input.

The bleed flow is specified as a massflow `W bleed` or a fraction `Bleed fraction` of the compressor flow (or no flow rate data for the last two options), depending on the bleed type. The `dH fraction` parameter indicates how much of the total compressor air enthalpy rise is applied to the bleed air. In other words, `dH fraction` indicates at about where between the high pressure exit (`dH fraction=1`) and lower pressure entry (`dH fraction=0`) the air is extracted. The higher `dH fraction`, the more power is taken from the compressor in terms of pressurized bleed air. Both `Bleed fraction` and `dH fraction` range between 0 and 1.
Note that depending in the Type selected, the `W bleed` or `Bleed fraction` columns can or cannot be edited. With the `Externally controlled` option, only `dH fraction` can be set.

Use the `Externally controlled` option in connection with a bleed control component if you want manual off-design control over the bleed flow rate.

Note that with bleed flow output data such as bleed mass flow rate, the parameter identifier is followed by the bleed nr. (e.g. `Wbld2_3` means bleed flow nr. 2 of component nr.3).

9 Errors

GSP contains extensive error handling routines providing information in the form of error messages.

Three types of error can be identified:

- *User input errors*
  User input errors are generated usually before (or sometimes after) a simulation run when GSP detects that input is invalid or conflicting with other input or that a model configuration is invalid.
- *Simulation errors*
  During simulation, errors may occur due to conditions such as inconsistent model configurations, thermo-dynamical limitations encountered beyond the operating envelope, unrealistic transient inputs and many more. Generally, errors result from non-realistic engine model operating conditions or component parameters.
- *Results processing errors*
  These type of errors may occur:
  - at the end of a simulation calculation during post-processing of simulation result data, calculation of derived parameters, or
  - during presentation and analysis of results in graphical or tabular format for example when parameters required for graphical presentation parameters are missing.

Errors can be generated by:

- the global GSP environment and/or the gas turbine model, or
- by the code for a specific component

The error messages are divided in two categories:

- Model specific error messages
- Component specific error messages

9.1 Model specific errors

9.1.1 Model specific errors

Model specific errors are generally a result of non-optimal iteration settings. When you get model error messages, first try using smaller steps between successive points calculated or specify less severe transient input changes.

The model specific errors are divided into:

1100 series - Demo specific errors
1200 series - Map specific errors
1300 series - I/O specific errors
1400 series - Iteration specific errors

9.1.2 Demo specific errors

The following demo specific errors can occur:

- 1101 - Saving *variable* is disabled in demo version
- 1102 - Evaluation period of GSP demo is expired
- 1103 - Custom/Application specific component library not enabled in GSP LE version

9.1.3 Map specific errors

The following map specific errors can occur:

- 1201 - Interpolation index out of range!
- 1202 - No Map Text File name specified in *component*
- 1203 - Field *field* for curve in map not found in table
- 1204 - Keyword "Reynolds:" not found in map file *filename*
- 1205 - Invalid first line map code string *string* in *map* (should be 99+blank)
- 1206 - Invalid Reynolds Index data line in *map*
- 1207 - Keyword *keyword* not found in *map*
- 1208 - Error interpolating in map *map* in *component*
- 1209 - Nr. of SurgeLine points does not correspond to nr. of speed lines in map file *map*
- 1210 - *filename* does not exist. Use Browse button on Component Map Tabsheet to find map
- 1211 - I/O error opening *filename.* Error code= *number*
- 1212 - Error reading map file *map.* Only *number* valid lines read
- 1213 - Error opening/reading map from file *map*
- 1214 - Error reading Map Data Row from file *map*
- 1215 - Error reading Map Data Row Block from file *map*
- 1216 - Could not call ReadMap_from_Textfile method

.

9.1.4 I/O specific errors

The following I/O specific errors can occur:

- 1301 - Cannot load custom package *package*
- 1302 - Cannot find/open model/table file *filename*
- 1303 - GSP model version ID: *version* not valid or invalid file *filename*
- 1304 - Cannot read GSP model file from file *filename*
- 1305 - Error reading *part* of Modelform *model*
- 1306 - Some fields not found while reading *component*…
- 1307 - Could not make output table *filename*
- 1308 - Could not output all data, table field(s) not found…
- 1309 - Error writing Model data lines
- 1310 - Field name *field* not found
- 1311 - Field name *field* exceeds maximum of *number* chars..
- 1312 - Field name *field* already exists ! Press OK to recalculate column
- 1313 - Need at least 2 fields in output file !
- 1314 - Could not sort Components, Component label format error
- 1315 - No activated input time series with 2 or more time steps specified
- 1316 - Start time *time* exceeds end time *time*
- 1317 - No steady state Data calculated since reset!
- 1318 - Cannot append table to itself
- 1319 - Class *class* does not exist in custom library package *package*
- 1320 - Field name *field* not valid ! Is already a GSP output DATA field
- 1321 - Model name changed from *old name* to *new name*. Must save with new Model file format
- 1322 - Component has no form !
- 1323 - Must calculate design point for States and Errors Report!
- 1324 - Exception error in Additional Output calculation in *component*
- 1325 - No Inlet component found!
- 1326 - Design point input changed ! Press Ok, if you want the model to be reset (and Design point will have to be recalculated). Press Cancel to go back to *component* edit form

- 1327 - No *table name* generated!
-

.

9.1.5 Iteration specific errors

The following iteration specific errors can occur:

- 1401 - No convergence in *procedure* in *component*
- 1402 - Nstates() <> Nerrors(). Model configuration inconsistent
- 1403 - Max. nr. of *number* Jacobian calculations encountered
- 1404 - Error in Broyden update: Division by zero
- 1405 - State restored to last converged operating point. Note that Control input and Flight Conditions may have to be reset/modified in order to prevent re-occurence of error !
- 1406 - Error in GetStatic iteration (choked flow case), contact NLR
- 1407 - *Error* exception in *component*
- 1408 - Negative Exit flow in vol. dyn. calculation in *component*
- 1409 - Error in vol. dyn. calculation in *component*
-

9.2 Component specific errors

9.2.1 Bleed specific errors

The following bleed specific errors can occur:

- 6201 - Bleed nr. *number* in Bleed list with type not set
- 6202 - Bleed nr. *number* does not exist
- 6203 - Sum of fractions used of bleed nr. *number* ... exceeds 1
- 6204 - Duplicate bleed nr. *number*

.

9.2.2 Component specific errors

Component specific errors generally result from non-realistic component parameters.

Component specific errors are divided into:

5100 series - Ambient/Flight conditions specific errors
5200 series - Inlet specific errors
5300 series - Fan specific errors
5400 series - Compressor specific errors
5500 series - Combustor specific errors
5600 series - Turbine specific errors
5700 series - Heat-Exchanger specific errors
5800 series - Mixer specific errors
5900 series - Exhaust specific errors
6000 series - Duct specific errors
6100 series - Controls specific errors

9.2.3 Flight conditions specific errors

The following ambient/flight condition specific errors can occur:

- 5101 - Invalid data in Ambient/Flight Conditions Trans.Grid! First time value must be 0
- 5102 - Invalid data in Ambient/Flight Conditions Trans.Grid! press Forms Cancel button to Cancel modification

9.2.4 Inlet specific errors

The following inlet specific errors can occur:

- 5201 - Both design RR and map design RR must be smaller than zero to enable map scaling!
- 5202 - Error in calculation of *component*

9.2.5 Fan specific errors

9.2.6 Compressor specific errors

For errors concerning bleed, see Bleed specific errors.

9.2.7 Combustor specific errors

The following combustor specific errors can occur:

- 5501 - ICAO emissions table Pt and Tt values zero or not descending in *component* emissions indices will not be calculated!
- 5502 - Error in ICAO/NLR Emission calculation
- 5503 - Total not 100%
- 5504 - Liquid water content must be zero for volume % specification
- 5505 - Warning: Equivalence ratio of combustion oxidant higher than one
- 5506 - Warning: Equivalence ratio too high for accurate Prompt-NOx prediction
- 5507 - (Design) Fuel temperature must be between 220 and 550 K
- 5508 - Sums of Fuel, Oxidator and Water inj. fractions must be 1
- 5509 - Fuel fraction must be larger than zero in 1st intersection
- 5510 - NO and N2O fractions cannot be used in fuel
- 5511 - Fuel pump shaft *number* does not exist in *component*
- 5512 - nO2in<0 in CombEquilibrium (combustion model)
- 5513 - O2 concentration too small
- 5514 - *Gas name* concentration *concentration*<0 in *component*
- 5515 - Composition sum *composition* in *component* not 1 !

9.2.8 Turbine specific errors

The following turbine specific errors can occur:

- 5601 - Shaftnr. *number* in *component* already driven by turbine
- 5602 - Free Power shaft nr. *number* already defined in *component*. Must be new shaft !
- 5603 - Undefined shaft nr. *number* in *component* must be free power shaft
- 5604 - User specified/Externally controlled speed shaft nr. *number* in *component* must either be free power shaft or drive compressor with user spec./extern.controlled speed
- 5605 - Design point shaft power absorbed cannot be 0 in *component*
- 5606 - No shaft nr. or suffix specified in *component*

9.2.9 Heat-Exchanger specific errors

The following heat-exchanger specific errors can occur:

- 5701 - Error calculating static conditions for Fundamental pressure loss in *component*. Check flow cross area: too small results in illegal supercritical flow !
- 5702 - Error calculating heat exchanger heat flow rate in *component*

9.2.10 Mixer specific errors

The following Mixer specific errors can occur:

- 5801 - Entry pressure conflict: Ps duct entry (*number* bar) larger than Pt core entry (*number* bar) (would cause reverse core flow !) in *component*

9.2.11 Exhaust specific errors

The following exhaust specific errors can occur:

- 5901 - Design point exhaust pressure ratio(=*number*) smaller than 1.0 in *component*

9.2.12 Duct specific errors

9.2.13 Controls specific errors

The following controls specific errors can occur:

- 6101 - Cannot get design *variable* from *component*. *Control* not properly linked to *component*
- 6102 - Shaft nr. *number* in *component* does not exist
- 6103 - Pb bleed nr. *number* in *component* does not exist
- 6104 - A transient calculation is not allowed if the *control* fully trimmed steady state option is checked

- 6105 - No shaft nr. or suffix specified in *component*
- 6106 - Combustor component *component* controlled by governor fuel control component *component* must have "Specify fuel" option set in design data

9.3 Errors

9.3.1 Error 0100

Error:
0100 - Cannot add component of type...

Cause:
A component is to be inserted into a configuration type not entitled to contain such a component.
A component can only be added in its designated configuration, or in ancestor configurations.

Action:
Please insert this component in a different configuration type.

The following types exist:
ctCfgNone | Component has no type
ctCfgProject | a project component type (for now unused)
ctCfgBaseCfg | insert in component in the base configuration model
ctCfgConfig | insert in component in configuration model node or an ancestor
ctCfgCase | insert in component in case model node or an ancestor

9.3.2 Error 0101

Error:
0101 - Cannot add component of type...
This component can only be inserted in...

Cause:
A component is to be inserted into a configuration type not entitled to contain such a component.
A component can only be added in its designated configuration, NOT in ancestor configurations.

Action:
Please insert this component in a different configuration type (as suggested in the dialog).

The following types exist:
ctCfgNone | Component has no type
ctCfgProject | a project component type (for now unused)
ctCfgBaseCfg | insert in component in the base configuration model
ctCfgConfig | insert in component in configuration model node or an ancestor
ctCfgCase | insert in component in case model node or an ancestor

9.3.3 Error 1101

Error:
1101 - Saving *variable* is disabled in demo version

Cause:
GSP demo version is limited

Action:
Register GSP (see NLR support)

9.3.4 Error 1102

Error:
1102 - Evaluation period of GSP demo is expired

Cause:
GSP demo version is limited

Action:
Register GSP (see NLR support)

9.3.5 Error 1103

Error:
1103 - Custom/Application specific component library not enabled in GSP demo version

Cause:
GSP demo version is limited

Action:
Register GSP (see NLR support)

9.3.6 Error 1201

Error:
1201 - Interpolation index out of range!

Cause:
Map variables out of range

Action:
Decrease value of map variables

9.3.7 Error 1202

Error:
1202 - No Map Text File name specified in *component*

Cause:
No map file name specified

Action:
Specify map file in `map` tab sheet

9.3.8 Error 1203

Error:
1203 - Field *field* for curve in map not found in table

Cause:
Required output variable not selected

Action:
Check `Map operating curve par.s` in `Output` tab sheet

9.3.9 Error 1204

Error:
1204 - Keyword "Reynolds:" not found in map file *filename*

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.10 Error 1205

Error:
1205 - Invalid first line map code string *string* in *map* (should be 99+blank)

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.11 Error 1206

Error:
1206 - Invalid Reynolds Index data line in *map*

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.12 Error 1207

Error:
1207 - Keyword *keyword* not found in *map*

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.13 Error 1208

Error:
1208 - Error interpolating in map *map* in *component*

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.14 Error 1209

Error:
1209 - Nr. of SurgeLine points does not correspond to nr. of speed lines in map file *map*

Cause:
Ill-defined map format

Action:
Check map format (see Technical Manual)

9.3.15 Error 1210

Error:
1210 - *filename* does not exist. Use Browse button on Component Map Tabsheet to find map

Cause:
Map file read error

Action:
Check and alter map file location

9.3.16 Error 1211

Error:
1211 - I/O error opening *filename.* Error code= *number*

Cause:
Map file read error

Action:
Check and alter map file location

9.3.17 Error 1212

Error:
1212 - Error reading map file *map.* Only *number* valid lines read

Cause:
Map file read error

Action:
Check and alter map file location

9.3.18 Error 1213

Error:
1213 - Error opening/reading map from file *map*

Cause:
Map file read error

Action:
Check and alter map file location

9.3.19 Error 1214

Error:
1214 - Error reading Map Data Row from file *map*

Cause:
Map file read error

Action:
Check and alter map file location

9.3.20 Error 1215

Error:
1215 - Error reading Map Data Row Block from file *map*

Cause:
Map file read error

Action:
Check and alter map file location

9.3.21 Error 1216

Error:
1216 - Could not call ReadMap_from_Textfile method

Cause:
Map file read error

Action:
Check and alter map file location

9.3.22 Error 1301

Error:
1301 - Cannot load custom package *package*

Cause:
Custom package not found

Action:
Check if custom package is located in GSP directory
Restart GSP after custom package has been placed in GSP directory

9.3.23 Error 1302

Error:
1302 - Cannot find/open model/table file *filename*

Cause:
Model or table file not found when reopening

Action:
Check and alter location of model or table file

9.3.24 Error 1303

Error:
1303 - GSP model version ID: *version* not valid or invalid file *filename*

Cause:
New model opened with old GSP version

Action:
Upgrade GSP version (see see NLR support)

9.3.25 Error 1304

Error:
1304 - Cannot read GSP model file from file *filename*

Cause:
Model file read error

Action:
Check model file

9.3.26 Error 1305

Error:
1305 - Error reading *part* of Modelform *model*

Cause:
Model file read error

Action:
Check model file

9.3.27 Error 1306

Error:
1306 - Some fields not found while reading *component*…

Cause:
Old model opened with upgraded GSP version

Action:
Check and alter component data for new variables

9.3.28 Error 1307

Error:
1307 - Could not make output table *filename*

Cause:
Disk writing error

Action:
Check drive and check path setting

9.3.29 Error 1308

Error:
1308 - Could not output all data, table field(s) not found…

Cause:
Previously generated table does not hold new variable

Action:
Use `Save/New Tables` to recreate output

9.3.30 Error 1309

Cause:

Action:

9.3.31 Error 1310

Cause:

Action:

9.3.32 Error 1311

Error:
1311 - Field name *field* exceeds maximum of *number* chars..

Cause:
Too long a calculation field name

Action:
Use shorter name and use New Name field

9.3.33 Error 1312

Error:
1312 - Field name *field* already exists ! Press OK to recalculate column

Cause:
New calculation field exists

Action:
Press OK to overwrite data or select a different field name

9.3.34 Error 1313

Error:
1313 - Need at least 2 fields in output file !

Cause:
Graph requires more than 1 variables

Action:
Check multiple variables in Output tab sheet

9.3.35 Error 1314

Component label format error

Cause:

Action:

9.3.36 Error 1315

Error:
1315 - No activated input time series with 2 or more time steps specified

Cause:
Transient calculation not activated or design ill-defined

Action:
Check the `Transient input activated` checkbox and specify more than 2 timesteps

9.3.37 Error 1316

Error:
1316 - Start time *time* exceeds end time *time*

Cause:
Start time larger than end time

Action:
Check and alter start time and end time

9.3.38 Error 1317

Error:
1317 - No steady state Data calculated since reset!

Cause:
Data for report not available

Action:
Click `steady state` before clicking `Report`

9.3.39 Error 1318

Error:
1318 - Cannot append table to itself

Cause:
Illegal append action

Action:
Do not use the selected table for `Append`

9.3.40 Error 1319

Error:
1319 - Class *class* does not exist in custom library package *package*

Cause:
Custom package does not contain library form with components

Action:
Contact NLR

9.3.41 Error 1320

Error:
1320 - Field name *field* not valid ! Is already a GSP output DATA field

Cause:
Name for calculated column already exists as standard output field

Action:
Specify alternate name for calculated column

9.3.42 Error 1321

Error:
1321 - Model name changed from *old name* to *new name*. Must save with new Model file format!

Cause:
Old model with .MDLOLD extension renamed but not saved to .MDL file

Action:
Save model as .MDL file in following step

9.3.43 Error 1322

Error:
1322 - Component has no form !

Cause:
Component icon has no corresponding data specification form.

Action:
Restart GSP and reload model. If the problem re-occurs, contact NLR.

9.3.44 Error 1323

Error:
1323 - Must calculate design point for States and Errors Report!

Cause:
Status and errors are not defined yet since no design point is calculated yet,

Action:
Calculate design point first by clicking `Design`

9.3.45 Error 1324

Error:
1324 - Exception error in Additional Output calculation in *component*

Cause:
Error in calculation of additional output (not required in iteration for operating point)

Action:
Check and alter component data

9.3.46 Error 1325

Error:
1325 - No Inlet component found!

Cause:
Inlet component not present on model form. Assigning component numbers always starts with the inlet component.

Action:
Drag-and-drop an inlet component on the model form and specify it's data

9.3.47 Error 1326

Error:
1326 - Design point input changed ! Press Ok, if you want the model to be reset (and Design point will have to be recalculated). Press Cancel to go back to *component* edit form.

Cause:
Design point input has been changed by last action (for instance by altering bleed flow in compressor)

Action:
Pressing `OK` will reset the model requiring a desgin point recalculation, while pressing `Cancel` will go back to the edit form to enable correction of your last action

9.3.48 Error 1327

Error:
1327 - No *table name* generated!

Cause:
Opening a non-existing table is not possible

Action:
Check the table name or click St.St.Series or Transient to produce tables

9.3.49 Error 1328

Error:
1328 - The output name 'XXX' already exists,
(same output parameter selected in multiple components)!

where 'XXX' denotes an output parameter name, usually a shaft ID

Cause:
The output parameter has been specified multiple times (in different components)

Action:
Since version 11 it is not allowed/not recommended to specify certain parameters multiple times. A temperature can be set by a components exit station, or by the next component inlet station temperature option. GSP will automatically check and repair old models where options are defined multiple times for most options. An exception are the shaft speed options, these are not automatically updated and have to be done manually.

There is a simple solution to solve this. Actually this is a warning, and not an error, so the calculation will continue if you press the ignore button.

There are 2 ways to solve this warning from popping up:

1) Individual output tab sheet setting (the following describes the setting of the output options for shaft speed, but can be applied to other output parameters)
   Check the `Output` tab sheet of every component that is connected to the same shaft and be sure only one of these components has the `Shaft` options (check boxes) `N [rpm]` and `N [%]` checked:
   a - Open your model
   b - Double click first component connected to the first shaft
   c - Go to the `Output` tab sheet
   d - Put a check mark in `N [rpm]` and `N [%]` options and close the component input window with `OK`
   e - Double click next component connected to the shaft
   f - Go to the Output tab sheet
   g - Clear the check mark in `N [rpm]` and `N [%]` options and close the component input window with `OK`
   h - Proceed to step e until all components for that shaft have been set
   i - If applicable, double click first component connected to the next shaft and proceed to step c

2) Central output parameter setting: (preferred solution)
1 - Open your model
2 - Go to the menu item `Options` then select `Output...`
3 - Go to the `Components` tab sheet on the `Output Parameters` tab sheet
4 - Set the drop down selection box at the bottom to `All Turbo Components`
5 - Put a check mark in `N [rpm]` and `N [%]` options (if these are already checked please uncheck and check again to activate the `Apply Changes` button)
6 - Press the `Apply Changes` button (this will automatically set the correct settings per shaft)
7 - Close the output options window with `OK`

9.3.50 Error 1401

Error:
1401 - No convergence in *procedure* in *component*

Cause:
Ill-defined component data

Action:
Check and alter component data for *component*

9.3.51 Error 1402

Error:
1402 - Nstates() <> Nerrors(). Model configuration inconsistent

Cause:
Invalid model configuration

Action:
Check and alter model configuration

9.3.52 Error 1403

Error:
1403 - Max. nr. of *number* Jacobian calculations encountered

Cause:
No solution found

Action:
Check and alter component data
Change timestep
Change accuracy

9.3.53 Error 1404

Cause:
Division by zero

Action:
Contact NLR

9.3.54 Error 1405

Error:
1405 - State restored to last converged operating point. Note that Control input and Flight Conditions may have to be reset/modified in order to prevent re-occurence of error !

Cause:
Automatic reset to last convergence point for correct iteration

Action:
Check control input, flight conditions and component data to prevent re-occurence of error.

9.3.55 Error 1406

Error:
1406 - Error in GetStatic iteration (choked flow case), contact NLR

Cause:
Iteration error in exhaust component

Action:
Contact NLR

9.3.56 Error 1407

Error:
1407 - *Error* exception in *component*

Cause:
Undefined exception error has occurred

Action:
Contact NLR

9.3.57 Error 1408

Error:
1408 - Negative Exit flow in vol. dyn. calculation in *component*

Cause:
Calculated exit flow smaller than 0 during transient

Action:
Adjust effective volume or contact NLR

9.3.58 Error 1409

Error:
1409 - Error in vol. dyn. calculation in *component*

Cause:
Undefined error in volume dynamics calculation during transient

Action:
Adjust effective volume or contact NLR

9.3.59 Error 5101

Error:
5101 - Invalid data in Ambient/Flight Conditions Trans.Grid! First time value must be 0

Cause:
Ill-defined transient data

Action:
Check and alter transient component data in table in `Transient` tab sheet

9.3.60 Error 5102

Error:
5102 - Invalid data in Ambient/Flight Conditions Trans.Grid! press Forms Cancel button to Cancel modification

Cause:
Ill-defined transient data

Action:
Check and alter transient component data in table in `Transient` tab sheet

9.3.61 Error 5201

Error:
5201 - Both design RR and map design RR must be smaller than zero to enable map scaling!

Cause:
Design RR and map design RR too high

Action:
Lower `Design RR factor` in `Design` tab sheet and adjust map design RR

9.3.62 Error 5202

Error:
5202 - Error in calculation of *component*

Cause:
Ill-defined component data

Action:
Check and alter *component* data

9.3.63 Error 5501

Error:
5501 - ICAO emissions table Pt and Tt values zero or not descending in *component* emissions indices will not be calculated!

Cause:
Ill-defined ICAO table in combustor for `Interpolation in ICAO table` emission model

Action:
Check ICAO table in `ICAO table` tab sheet in combustor `Emissions` tab sheet

9.3.64 Error 5502

Error:
5502 - Error in ICAO/NLR Emission calculation

Cause:

Ill-defined ICAO table in combustor for `Interpolation in ICAO table` emission model

Action:
Check ICAO table in `ICAO table` tab sheet in combustor `Emissions` tab sheet

9.3.65 Error 5503

Error:
5503 - Total not 100%

Cause:
Ill-defined `Composed fuel specification`

Action:
Check `Composed fuel specification` table in combustor `Design fuel` tab sheet

9.3.66 Error 5504

Error:
5504 - Liquid water content must be zero for volume % specification

Cause:

Action:
Select `Mass %` for `Composed fuel specification` in combustor `Design fuel tab sheet`

9.3.67 Error 5505

Error:
5505 - Warning: Equivalence ratio of combustion oxidant higher than one

Cause:
NOx formation cannot be predicted if the flow is already fuel-rich without the fuel added

Action:

9.3.68 Error 5506

Error:
5506 - Warning: Equivalence ratio too high for accurate Prompt-NOx prediction

Cause:
NOx formation cannot be predicted if the flow is already fuel-rich without the fuel added

Action:

9.3.69 Error 5507

Error:
5507 - (Design) Fuel temperature must be between 220 and 550 K

Cause:
(Design) Fuel temperature out of limits

Action:
Alter `Fuel temperature` in `Standard fuel specification` in `(Design) Fuel` tab sheet

9.3.70 Error 5508

Error:
5508 - Sums of Fuel, Oxidator and Water inj. fractions must be 1

Cause:
Ill-defined fractions in multi-reactor emission model table

Action:
Check fractions in table in `Multi-reactor` tab sheet in `Emissions` tab sheet

9.3.71 Error 5509

Error:
5509 - Fuel fraction must be larger than zero in 1st intersection

Cause:
Ill-defined fuel fraction in first line in multi-reactor emission model table

Action:
Check fuel fraction in first row of table in `Multi-reactor` tab sheet in `Emissions` tab sheet

9.3.72 Error 5510

Error:
5510 - NO and N2O fractions cannot be used in fuel

Cause:
NO and N2O fractions cannot be used in User specified (design) fuel

Action:
Set NO and N2O fractions to zero in `Composed fuel specification` table in `(Design) Fuel` tab sheet

9.3.73 Error 5511

Error:
5511 - Fuel pump shaft *number* does not exist in *component*

Cause:
Shaft nr. is not specfied for fuel pump

Action:
Set shaft nr./suffix for fuel pump in `Fuel Pump` tab sheet

9.3.74 Error 5512

Error:
5512 - nO2in<0 in CombEquilibrium (combustion model)

Cause:
Input number of O2 moles smaller than 0

Action:

Contact NLR

9.3.75 Error 5513

Error:
5513 - O2 concentration too small

Cause:
Fuel-to-air ratio too high or accuracy too low

Action:
Decrease fuel flow or increase air flow, or increase accuracy.

9.3.76 Error 5514

Error:
5514 - *Gas name* concentration *concentration*<0 in *component*

Cause:
Ill-defined fuel composition

Action:
Change fuel composition or increase accuracy

9.3.77 Error 5515

Error:
5515 - Composition sum *composition* in *component* not 1 !

Cause:
Ill-defined fuel composition

Action:
Change fuel composition or increase accuracy

9.3.78 Error 5516

Error:
5516 - Error in calc method of Multi Reactor component

Cause:

Action:
contact NLR

9.3.79 Error 5601

Error:
5601 - Shaftnr. *number* in *component* already driven by turbine

Cause:
Shaftnumber is already defined for shaft

Action:
Change shaftnumber in *component*

9.3.80 Error 5602

Error:
5602 - Free Power shaft nr. *number* already defined in *component*. Must be new shaft !

Cause:
Shaftnumber is already defined for shaft

Action:
Change shaftnumber in *component*

9.3.81 Error 5603

Error:
5603 - Undefined shaft nr. *number* in *component* must be free power shaft

Cause:
Shaftnumber is defined without turbine

Action:
Check `Free Power Turbine` in turbine `General` tab sheet

9.3.82 Error 5604

Error:
5604 - User specified/Externally controlled speed shaft nr. *number* in *component* must either be free power shaft or drive compressor with user spec./extern.controlled speed

Cause:

Action:

9.3.83 Error 5605

Error:
5605 - Design point shaft power absorbed cannot be 0 in *component*

Cause:
No design external load specified

Action:
Specify design external load in turbine *Design* tab sheet

9.3.84 Error 5606

Error:
5606 - No shaft nr. or suffix specified in *component*

Cause:
No shaft nr. defined

Action:
Specify shaft nr. in turbine *General* tab sheet

9.3.85 Error 5701

Error:
5701 - Error calculating static conditions for Fundamental pressure loss in *component*. Check flow cross area: too small results in illegal supercritical flow !

Cause:
Flow cross area too small

Action:
Check and alter `Flow cross area` in both `Flow pressure loss` tabsheets

9.3.86 Error 5702

Error:
5702 - Error calculating heat exchanger heat flow rate in *component*

Cause:
Ill-defined component data

Action:
Check and alter `Design` component data

9.3.87 Error 5801

Error:
5801 - Entry pressure conflict: Ps duct entry (*number* bar) larger than Pt core entry (*number* bar) (would cause reverse core flow !) in *component*

Cause:
Ill-defined component data

Action:
Check and alter `Duct entry area` and `Core entry area` in `Model Data` in `General` tab sheet

9.3.88 Error 5901

Error:
5901 - Design point exhaust pressure ratio(=*number*) smaller than 1.0 in *component*

Cause:
Pressure ratio can not be smaller than 1

Action:
Check and alter model data

9.3.89 Error 6101

Error:
6101 - Cannot get design *variable* from *component*. *Control* not properly linked to *component*

Cause:
Control not linked to *component*

Action:
Link *control* to *component*

9.3.90 Error 6102

Error:
6102 - Shaft nr. *number* in *component* does not exist

Cause:
Incorrect shaft nr. specified

Action:
Check and alter `shaft nr.` in `General` tab sheet

9.3.91 Error 6103

Error:
6103 - Pb bleed nr. *number* in *component* does not exist

Cause:
Incorrect Pb bleed nr. specified

Action:
Check and alter `Pb bleed nr.` in `General` tab sheet

9.3.92 Error 6104

Error:
6104 - A transient calculation is not allowed if the *control* fully trimmed steady state option is checked

Cause:
The *control* `fully trimmed steady state` option is checked in the *control*'s `Schedules` tab sheet.

Action:
Uncheck the `fully trimmed steady state` option in the `Schedules` tab sheet.

Background:
When the option is checked, the steady state point determined is the actual fully trimmed steady state point with the effect of the control's integrator in time. This option therefore includes both the determination of the (untrimmed) steady state point as well as the transient run required to reach the fully trimmed situation.
Normally, a steady state point does not coincide with the fully trimmed steady state point if a control is present. Instead, the fully trimmed point must be achieved by running a transient until the engine is stabilized.

9.3.93 Error 6105

Error:
6105 - No shaft nr. or suffix specified in *component*

Cause:
No shaft nr. defined

Action:
Specify shaft nr. in control *General* tab sheet

9.3.94 Error 6106

Error:
6106 - Combustor component *component* controlled by governor fuel control component *component* must have "Specify fuel" option set in design data

Cause:
Combustor component has specification of `Exit temperature` or `Fuel-Air ratio` instead of `Fuel flow` in `Specify` radiogroup on `Design` tab sheet

Action:
Set specification to `Fuel flow` in `Specify` radiogroup on combustor `Design` tab sheet

9.3.95 Error 6201

Error:
6201 - Bleed nr. *number* in Bleed list with type not set

Cause:
No bleed type is set for specific bleed flow in compressor component

Action:
Specify bleed type in compressor `Bleed` tab sheet

9.3.96 Error 6202

Error:
6202 - Bleed nr. *number* does not exist

Cause:
A component uses a bleed flow which is not specified in the compressor component

Action:
Specify a new bleed flow with *number* in the compressor `Bleed` tab sheet or delete the bleed flow number from the component

9.3.97 Error 6203

Error:
6203 - Sum of fractions used of bleed nr. *number* ... exceeds 1

Cause:
Total bleed flow used in components is larger than bleed flow available from compressor for bleed nr. *number*

Action:
Decrease bleed flow fractions used in components or increase bleed flow available from compressor for bleed nr. *number*

9.3.98 Error 6204

Error:
6204 - Duplicate bleed nr. *number*

Cause:
Compressor component contains duplicate bleed numbers

Action:
Check and correct the bleed numbers in the compressor `Bleed` tab sheet.

9.4 Help not implemented

In case the following error is encountered, please contact the GSP Development Team for more information.

"Help for this item is not yet implemented."

Please report the related item and GSP version number, which can be found by clicking `Help|About` in the main program window, or alternately right-clicking the file icon of the `GSP.exe` file (located in the installation directory, e.g. ..`\Program Files\NLR\GSP\`), selecting `Properties` and looking in the `Version` tab sheet.

10 Registration & Support

10.1 Contact details

For questions, problems or support please contact NLR using the special feedback forms on the GSP site at www.gspteam.com or directly through info@gspteam.com. Registered customers can contact the GSP team using the support@gspteam.com e-mail address.

10.2 Registration

GSP is protected with a registration key. An unregistered version will have limited capabilities, i.e. saving of many non-standard components (see the feature matrix). Registration of GSP gives you the benefit of NLR support in the form of answering questions, assisting in solving problems and development of new models according to the license agreement.

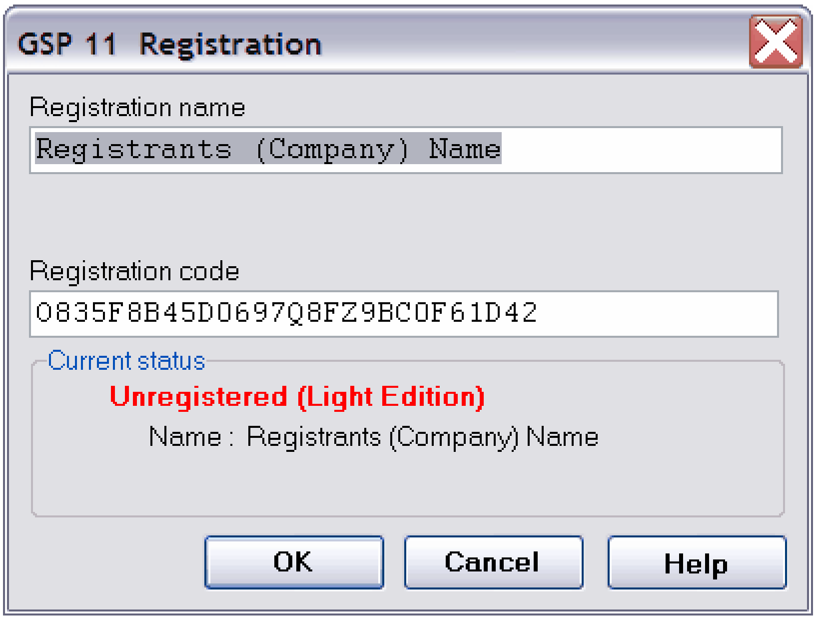
For license and registration information, please navigate to the licensing and order web pages. If there are further questions please do not hesitate to contact us.

The registration code is coupled to a specific registrant name. The registration information if supplied through e-mail upon payment delivery. Copy both the registrant name and the registration key into the input boxes of the registration window.

10.2.1 Registration window

Open the registration window by selecting the Registration menu item from the GSP main window `Help` menu.

Copy the registrants name supplied in the registration e-mail into the `Registration name` input field and copy the registration key into the `Registration code` input field. "Copy-paste" is preferred above typing the name and code to prevent typing errors (e.g. zeros, 0, and capital letters O look very similar). Press OK to register the application.



10.3 Support from NLR

The National Aerospace Laboratory NLR only supports *registered* users of GSP by answering questions, assisting in solving problems, development of new models or development of customized components, according to the limitations in the applicable license agreement. Optional support can be purchased.

Since GSP is continuously being extended and updated following user suggestions, we appreciate comments and bug reports of both users of the standard version as well as registered users.

When contacting for a support question, please report

- the GSP version number
  The version can be found by clicking `Help|About` in the main program window, or alternately right-clicking the file icon of the `GSP.exe` file typically located in the installation directory, e.g. `.\Program Files\NLR\GSP\`, selecting `Properties` and looking in the `Version` tab sheet), and
- describe the problem, or systematically describe how the problem arises
- if possible do send the model/project so that is is easier for the GSP development team to reproduce the error

A list of Frequently Asked Questions can be found on the GSP home site at www.gspteam.com.

11 References

1. Visser W.P.J., Broomhead M.J., 2000, ***GSP, A Generic Object-Oriented Gas Turbine Simulation Environment***, ASME 2000-GT-0002, ASME IGTI Turbo Expo conference, Munich, 2000

2. Tinga T., Visser W.P.J., de Wolf W.B., and Broomhead M.J., 2000, ***Integrated Lifing Analysis Tool for Gas Turbine Components***, ASME 2000-GT-0646, ASME IGTI Turbo Expo conference, Munich 2000

3. Visser W.P.J., 1995, ***Gas Turbine Simulation at NLR, Making it REAL***, CEAS Symposium on Simulation Technology (paper MOD05), Delft, the Netherlands

4. Visser W.P.J., Kluiters S.C.M., 1999, ***Modeling the Effects of Operating Conditions and Alternative Fuels on Gas Turbine Performance and Emissions***, NLR Technical Publication NLR-TP-98629 or Research and Technology Organization RTO-MP-14

5. Visser W.P.J., Kogenhop O., Oostveen. M., 2004, ***A Generic Approach for Gas Turbine Adaptive Modeling***, ASME GT2004-53721, ASME IGTI Turbo Expo conference, Vienna 2004 and ASME Journal of Engineering for Gas Turbines and Power, Volume 128 2006

6. Visser W.P.J., Oostveen M., Pieters H., Dorp E. van, 2006 ***Experience with GSP as a Gas Path Analysis Tool***, ASME GT2006-90904, ASME IGTI Turbo Expo conference, Barcelona 2006

7. Michel L. Verbist, Wilfried P.J. Visser, Jos P. van Buijtenen, Rob Duivis, ***Gas Path Analysis on KLM in-Flight Engine Data***, ASME paper GT2011-466257, presented at the ASME TURBO EXPO 2011, 6-10 June, Vancouver, Canada

8. Verbist, M.L., Visser, W.P.J., van Buijtenen, J.P., Duivis, ***Model-based gas turbine diagnostics at KLM Engine Services***, XX ISABE Conference, 2011, Gothenburg, Sweden, ISABE-2011-1807

9. Verbist, M.L., Visser, W.P.J., Pecnik, R., van Buijtenen, J.P, ***Component Map Tuning Procedure using Adaptive Modeling***. ASME Turbo Expo 2012, Copenhagen, Denmark, GT2012-69688

10. ***Manual of the ICAO standard atmosphere extended to 80 kilometres (262 500 feet)***, ICAO 7488/3, 1993

11. Kurzke J., 1996, ***How to get Component Maps for Aircraft Gas Turbine Performance Calculations***, ASME 96-GT-164

12. Kurzke J., 2005, ***How to Create a Performance Model of a Gas Turbine From a Limited Amount of Information***, ASME GT 2005-68537

13. RTO TECHNICAL REPORT RTO-TR-044
"Performance Prediction and Simulation of Gas Turbine Engine Operation"

14. RTO TECHNICAL REPORT TR-AVT-036

Index

- . -

- 1 -

- 2 -

- A -

- B -

- C -

- D -

- E -

- F -

- G -

- H -

- I -

- L -

- M -

- N -

- O -

- P -

- R -

- S -

GSP

Gas turbine Simulation Program

www.GSPteam.com