Dedicated to innovation in aerospace

NLR-TP-2020-010 | February 2020

Optimisation methodology for integrated equipment installation in new engine architecture nacelles

CUSTOMER: Royal Netherlands Aerospace Centre


© Eric Drouin / Safran

NLR – Royal Netherlands Aerospace Centre

Optimisation methodology for integrated equipment installation in new engine architecture nacelles

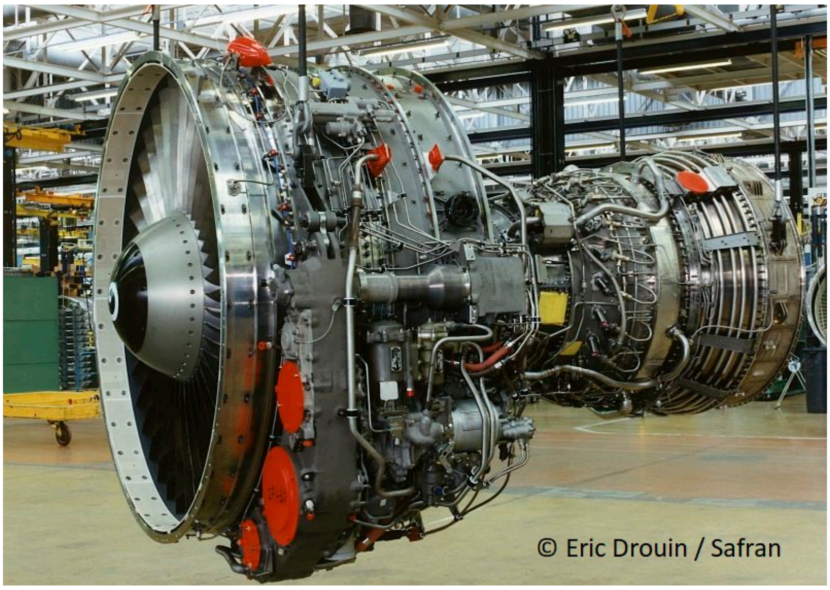


Illustration of equipment installation on a CFM56-3 engine

REPORT NUMBER
NLR-TP-2020-010

AUTHOR(S)
W.J. Vankan
R. Maas
V. Peyron

REPORT CLASSIFICATION
UNCLASSIFIED

DATE
February 2020

KNOWLEDGE AREA(S)
Aerospace Collaborative Engineering and Design

DESCRIPTOR(S)
Optimisation
space allocation
graph theory
engine equipment
ultra-high bypass ratio

Problem area

With the development of new aero engines, such as the Ultra-High Bypass Ratio (UHBR) engine and open-rotor solutions, advances in propulsion and fuel efficiency can be made through increasing fan diameters and higher bypass-ratios. Several enhancements to the overarching Integrated Power Plant System (IPPS) design are needed, in particular a shorter and slimmer nacelle. This will provide less space for the engine systems installation and for ventilation solutions. Moreover, higher operating temperatures and accessibility for maintenance further complicate the feasible arrangement of all the required engine equipment. To address these challenges, better ways of developing integrated solutions for the IPPS are required. Improved technologies that allow the complete system to be developed and trade studies to be assessed have been investigated in the EU H2020 research project NIPSE. These technologies are aimed at finding solutions to the challenges created by next generation of UHBR turbofan aero engines in terms of engine equipment integration.

Description of work

In the NIPSE project, new optimisation methodologies aimed at optimising in 3D space have been investigated. This work has generated an optimisation tool (NEAT: Next generation engine Equipment Allocation Tool) that considers the placement of equipment as well as the automated routing of pipes, ducts and harnesses between the equipment in the nacelle. More specifically, the implementation of the advanced optimisation methodologies into an efficient software framework is considered, with focus on the integration with the CAD systems (such as CATIA) that are typically used in the industrial aero engine and nacelle design process.

Results and conclusions

It is demonstrated for two design cases from the NIPSE project that both efficiency and flexibility are ensured in this integration process. A relatively simple electric-hydraulic case study of the installation of some electric generator and heat exchanger equipment in a representative nacelle interior geometry has demonstrated the main functionality and the basic operation of the software tool NEAT. Also another more advanced case study of the so-called ETRAS (electric thrust reverser actuation system) with eight equipment and simple box- and cylinder-shaped equipment geometries was considered. It was shown for the optimal dressings for the ETRAS case study that the main constraints that were considered in this case study were correctly taken into account. It was also found that with the NEAT tool the optimization of the dressings in the installation of the ETRAS case study required about 8min on a standard PC, which is about 22min (75%) less than the manual creation directly in CATIA of the same 13 dressings. It was concluded from the case studies that the automated routing process allows for quick testing and comparing of many configurations (i.e. modifying the location of equipment), which is something very valuable in the early design phases of an IPPS in order to identify the most optimised configuration in terms of weight, cost and maintenance.

Applicability

Besides the optimisation methodology, also the software implementation is presented, in particular the software tool NEAT and its application to aero engine equipment installation design cases. The methodology is very flexible and well-suited for other industrial applications like harnesses routing and design for whole aircraft, fuselage or wing.

GENERAL NOTE

This report is based on an article published in Journal of Aerospace Engineering, Proceedings of the Institution of Mechanical Engineers Part G [PIG], Article first published online: December 21, 2019.

NLR-TP-2020-010 | February 2020

Optimisation methodology for integrated equipment installation in new engine architecture nacelles

CUSTOMER: **Royal Netherlands Aerospace Centre**

AUTHOR(S):

W.J. Vankan	NLR
R. Maas	NLR
V. Peyron	Safran Nacelles

NLR – Royal Netherlands Aerospace Centre

This report is based on an article published in Journal of Aerospace Engineering, Proceedings of the Institution of Mechanical Engineers Part G [PIG], Article first published online: December 21, 2019.

CUSTOMER	**Royal Netherlands Aerospace Centre**
CONTRACT NUMBER	**636218**
OWNER	**NLR**
DIVISION NLR	**Aerospace Vehicles**
DISTRIBUTION	**Unlimited**
CLASSIFICATION OF TITLE	**UNCLASSIFIED**

APPROVED BY:		**Date**
AUTHOR	W.J. Vankan	10-01-2020
REVIEWER	J. Kos	10-01-2020
MANAGING DEPARTMENT	A.A. ten Dam	20-02-2020

Contents

Abstract 4

1 Introduction 5

2 UHBR engine and the challenge in the nacelle 6

3 Optimization methodology for integrated equipment installation 8

4 Equipment placement and routing in the graph 11

5 Aero-engine equipment, connectors and dressings 12

6 Graph-based routing 13

7 Software tool development 15

8 (Application case studies) Basic use case on electric-hydraulic components 16

9 Advanced use case on NIPSE-ETRAS components 22

10 Conclusions and further work 27

Acknowledgements 28

References 28

Optimisation methodology for integrated equipment installation in new engine architecture nacelles (old: Integrated equipment installation and optimisation for new engine architecture nacelles)

Wilhelmus J. Vankan, Robert Maas*, Vincent Peyron***

** Netherlands Aerospace Centre NLR, Dept. Collaborative Engineering Systems, Aerospace Vehicles Division*

A. Fokkerweg 2, 1059 CM Amsterdam, the Netherlands; jos.vankan@nlr.nl

*** Safran Nacelles, Direction Technique, France*

Abstract

The development of new higher efficiency turbofan aero engines requires several design enhancements that typically result in shorter and slimmer nacelles. Consequently these engines provide less space for the engine systems installation and for maintenance accessibility. In the NIPSE project [1], optimisation methodologies are being investigated and developed for the integrated installation of equipment into the restricted volume of new architecture engines' nacelles. The underlying optimisation methodology is built on a graph based approach involving efficient routing algorithms. Besides the methodology, also the software implementation and its application to engine equipment installation design cases are presented in this paper.

Keywords

Optimisation, space allocation, graph theory, engine equipment, UHBR

1. Introduction

With the development of new turbofan aero engines, such as the Ultra-High Bypass Ratio (UHBR) engine and open-rotor solutions, advances in propulsion and fuel-efficiency can be made through increasing fan diameters and higher engines' bypass-ratios [2]. For this, further examination of the integration of the engine into the aircraft is needed. Several enhancements to the overarching integrated power plant system (IPPS) design are needed to achieve the envisaged higher efficiency of these engine architectures, in particular a shorter and slimmer nacelle. This will provide less space for the engine systems installation and for ventilation solutions. Moreover, higher operating temperatures and accessibility for maintenance further complicate the feasible arrangement of all the required engine equipment.

To address these challenges, better ways of developing integration solutions for the IPPS are required. Optimisation methodologies and toolsets that allow the complete system to be developed and trade studies to be assessed have been investigated in the European Union Horizon 2020 research project NIPSE (Novel Integration of Powerplant System Equipment) [1]. These methodologies and toolsets consider the IPPS to be optimised as a whole, with best solutions for weight and drag easily identified very quickly during the development process. This optimisation approach considers the placement of equipment as well as the routing of ducts and harnesses between these equipment. This not only provides a potentially better answer than current methodologies used in industry, but the automation aspect provides solutions faster allowing the development time of the IPPS to be reduced. This development is achieved by applying different optimisation methodologies to key variables in order to identify the best overall methodologies to achieve the best solution in the shortest time.

Optimisation algorithms for routing problems have been extensively investigated in the past, e.g. for optimum route planning in transportation problems. So called graph algorithms have proven very successful for that. Early developments are for example the well-known Dijkstra's algorithm [3] and the A* algorithm [4] and many subsequent variants and implementations thereof [5], all of which address the basic problem of the calculation of the shortest path in a graph of interconnected nodes between two given points, assigned as source node and destination node. More specifically, graph based approaches for system installation problems have also been investigated. For example [6] presents the investigation of machine-executed compilation of graph-based design languages to efficiently address topological and parametrical design problems, with application to aircraft cabin system design.

Typically, for example in harnesses or pipework routing, important criteria for the routed path are not only total path length, but also other cost factors that may depend on local conditions in the routing space, like installation aspects (clamps, insulation, thermal protection etc.) and segment bend radius of local curvature, as has been investigated for example in [7].

The present paper describes the design and optimisation methodology that is developed for the integrated equipment installation into the restricted volume of new architecture aero engines' nacelles. The focus in this paper is on highly efficient and flexible placement of equipment and automated routing of pipes, ducts and harnesses. The implementation of the methodology aims for easy user interaction, allowing design engineers to quickly make changes to equipment installations taking into account the relevant installation rules for example for clearances, accessibility and temperature constraints. This optimisation methodology is built on a graph based approach involving efficient routing algorithms that rely on variants of Dijkstra's algorithm. Besides the methodology, also the software implementation is presented, in particular the software tool NEAT (Next generation engine Equipment Allocation Tool) and its application to aero engine equipment installation design cases.

2. UHBR engine and the challenge in the nacelle

Over the last decades, the bypass ratio and hence the propulsive efficiency of aero engines has been continuously increased. Current state of the art turbofan engines have bypass ratios of about 10 and overall pressure ratios of about 40. Further developments in aero engine technology are typically aiming for increasing the overall pressure ratios to improve thermal efficiency and for higher bypass ratios to improve propulsive efficiency [8]. These technology developments, such as optimised fans, gearbox cooling and variable area fan nozzle [9], are investigated in dedicated research programs for application on advanced UHBR turbofan engines. These UHBR engines have bypass ratios between 12 and 20 and overall pressure ratios between 50 and 70, with potential thermal efficiency beyond 60% and propulsive efficiencies beyond 80%. Investigations in several recent European research programs have focussed for example on improvements in new engine components and subsystems [10],[11], improvement of the core-engine thermal efficiency [12],[13] and on the low pressure system [2] of UHBR engines.

New turbofan aero engines, like the UHBR engine, require smaller and thinner nacelles. As a result, the engine equipment shall be installed in more confined space with higher operating temperatures. An indicative illustration of this spatial reduction of a UHBR nacelle in comparison with a proportionally scaled current state-of-the-art nacelle is given by the red (scaled current) and green (UHBR) areas in the figure 1 below.

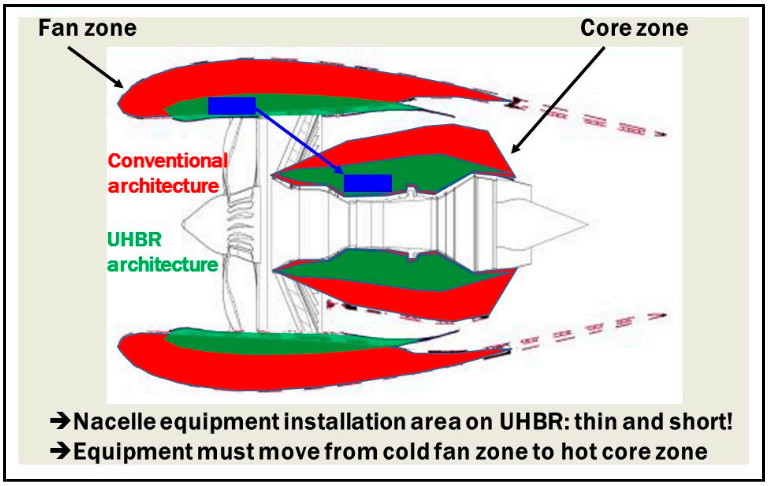


Figure 1: Illustration of the smaller and thinner nacelles typically required for UHBR engines.

The equipment installation in today’s conventional aero engine nacelles is already very complex due to confined space and difficult shapes, as illustrated for example for a CFM56-3 engine in figure 2.

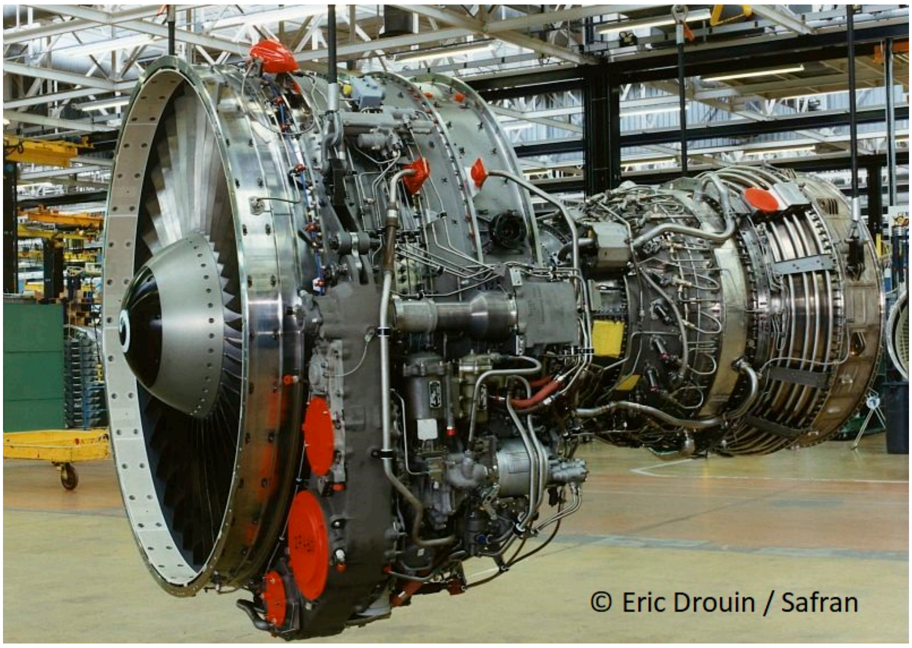

© Eric Drouin / Safran

Figure 2: Illustration of equipment installation on a CFM56-3 engine. [14]

In a typical engine system equipment package, there are around 40 different sets of equipment, including Engine Build Unit (EBU) and Buyer Furnished Equipment (BFE) which link the engine to the aircraft. Of these 40, around 20 are considered key equipment and account for about 80% of the weight (for example generator, fuel pump, lubrication unit, fuel management unit, Engine Electronic Controller (EEC), actuators, heat exchangers). 90% of this equipment is classed as Line Replaceable Units (LRU) and must be replaceable in the field. Of the remaining weight, 50% is ducting and pipework for hydraulic and pneumatic systems, 20% is electrical wiring harnesses with the remaining weight necessary to mount the systems. This system complexity leads to difficult challenges for equipment placement, for example in relation to accessibility for routine maintenance and/or replacement of equipment. Equipment installation for future engines must provide solutions that do not reduce the current maintenance access to key equipment.

3. Optimization methodology for integrated equipment installation

In the optimization methodology the confined space in the nacelle is modelled as a discretized three dimensional (3D) volume. This will be first explained on a strongly simplified example of this confined space in the nacelle, identified here as the keep-in-zone (KIZ) for the engine equipment. A much more detailed KIZ will be applied in the use case studies presented in the sections 8 and 9 below. The KIZ globally comprises some interconnected compartments: mainly the fan zone, the bifurcation zone and the core zone, as illustrated in figure 3.

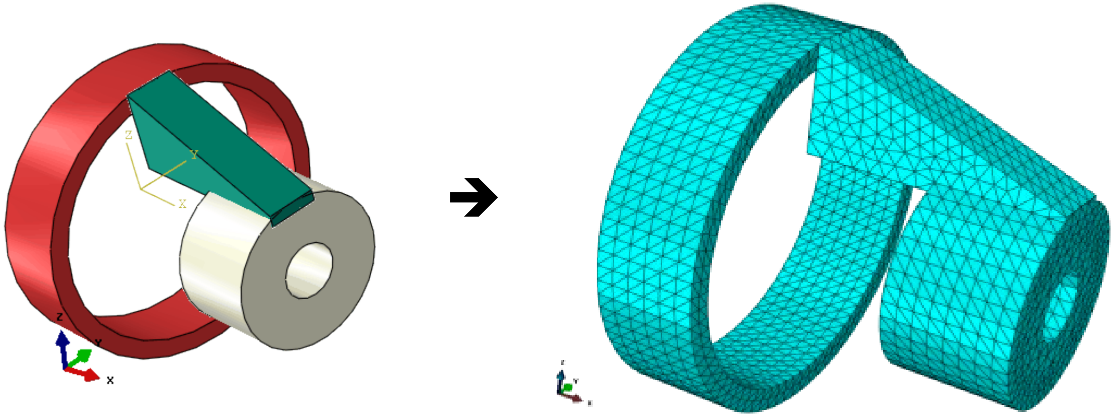


Figure 3: Illustration of the interconnected compartments (fan zone in red, bifurcation in green, core zone in grey) in the simplified KIZ as defined in a 3D CAD model (left), and its discretised representation by unstructured triangulated surfaces (right).

This KIZ is typically defined in three dimensional (3D) computer aided design (CAD) models by aero engine OEMs (original equipment manufacturers) and nacelle manufacturers. To translate such 3D CAD models of the KIZ into models that can be used for installation and routing optimisation, these CAD models can be simplified and discretised typically into unstructured triangulated surface representation as for example in STL (stereo-lithography) file format. Such STL file then contains the (approximate) representation of the outer surface of the complete KIZ's 3D geometry. The STL surface representation can be used to create a graph that represents a large set of small 3D cells that approximately fill the complete volume of the KIZ. Typically cubic cells or other hexahedral shapes are used, but potentially also cells with other shapes like tetrahedral or pentahedral might be used. Such a KIZ-graph can for example be achieved with a set of cube-shaped cells defined in a Cartesian grid, as illustrated in figure 4.

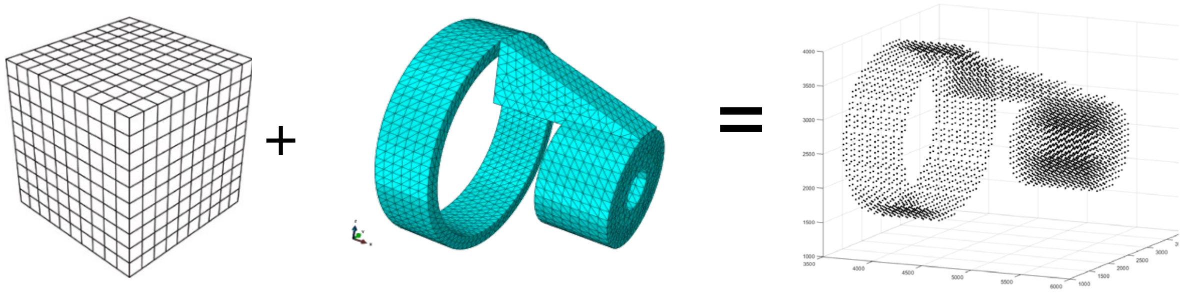
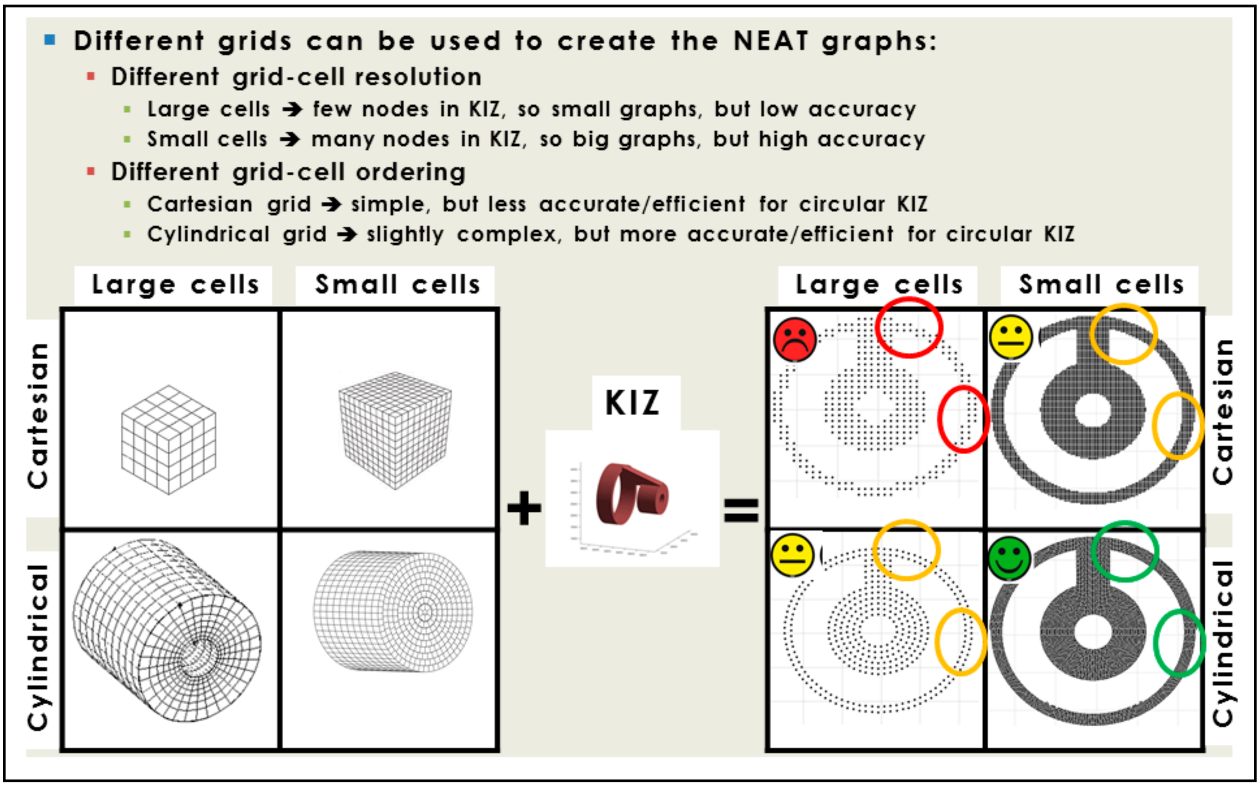


Figure 4: Illustration of the creation of a KIZ-graph, in this case based on a very coarse grid of cells. The KIZ-graph is created based on a set of cubes defined in a Cartesian grid (left). From this grid, the cubes are selected that are inside the KIZ triangulated surface (middle).These selected cubes are then used for the KIZ-graph, i.e. these graph-nodes represent the centres of the selected cubes (the black dots in the picture on the right).

It should be noted that the graph essentially represents only the set of indexed nodes, e.g. representing the centres of the 3D cells inside the KIZ, and the desired connections between the nodes, e.g. for each node the connection to each of its neighbouring nodes. Thus the nodes can be linked to 3D coordinates that represent locations in the KIZ, and the node-connections can be linked to the 3D Euclidian distances between the connected nodes. This allows for a highly efficient data structure on which graph based routing algorithms can be applied for highly efficient calculations of shortest path between arbitrary pairs of nodes. Typically the core data structure requires only $(n_{c,i} + 1)$ integers per node, where $n_{c,i}$ is the number of connections of node i, and one real number l_j per connection representing the length of connection j.

All sorts of manipulations can be performed on the graph, for example the exclusion of a node from the graph once this node has been assigned to a routed path, and as such represents an obstacle for subsequently routed paths. Similarly nodes can be excluded from the graph if they are assigned to an equipment, i.e. if an equipment (for example an electric component box) is defined in a certain location in the KIZ. Of course many other types of manipulations of the graph and its further processing are possible, for example by applying “weights” or “penalty factors” to certain preferred of disfavoured node connections due to certain routing directions or temperature conditions in the KIZ.

Various settings for the creation of the graph can be defined. Important choices are for example whether a 3D Cartesian or cylindrical grid of cells is used to create the graph. For the considered aero-engine KIZ, which is more or less cylindrical, a cylindrical grid of cells is more efficient for conformal mapping of the grid cells to the KIZ surface. Also, obviously, the cell size of the grid is important for the accuracy of the mapping of the grid cells to the KIZ geometry, see figure 5.



- Different grids can be used to create the NEAT graphs:
  - Different grid-cell resolution
    - Large cells ➔ few nodes in KIZ, so small graphs, but low accuracy
    - Small cells ➔ many nodes in KIZ, so big graphs, but high accuracy
  - Different grid-cell ordering
    - Cartesian grid ➔ simple, but less accurate/efficient for circular KIZ
    - Cylindrical grid ➔ slightly complex, but more accurate/efficient for circular KIZ

Figure 5: Illustration of the effects of different choices for the grid shape and grid cell size on the accuracy of the created KIZ-graph.

The grid cell size is also important for the representation of the size of the routed components. Typically the routing can be most efficiently performed if these sizes are more or less equal. For example, the routing of a harness with a 10 mm diameter is most efficiently achieved if a 10 mm grid cell size is used. Hence, several graphs based on different grid cell sizes can be applied for quick routing of differently sized components. Alternatively, for more accurate geometric representation and detailed routing, small grid size can also be used for the routing of larger diameter components.

4. Equipment placement and routing in the graph

With the KIZ-graph available, equipment can be easily inserted in the graph. Equipment exclude from the graph certain groups of nodes that are located in the KIZ space that is occupied by the equipment. Equipment typically have several connector-nodes, i.e. nodes to which pipes, ducts or harnesses can be connected. The pipes, ducts or harnesses will be further referred to as “dressings”. As such, connections between equipment can be defined. Also connections between equipment and/or arbitrary locations in the KIZ (e.g. harness connectors at the KIZ surface) can be made, as is illustrated in figure 6. With the graph algorithms, optimised routings of dressings can be very efficiently determined from the KIZ-graph and various constraints or penalties can be incorporated in these routing optimisations. The resulting routed dressing as such occupies the KIZ space of the grid cells that correspond to the sequence of graph nodes that were determined for the dressing’s route. If the grid cell size (typically: cube edge length) more or less corresponds to the size of the dressing (typically: the diameter), then the routing can be efficiently done with the graph that is based on that grid cells size.

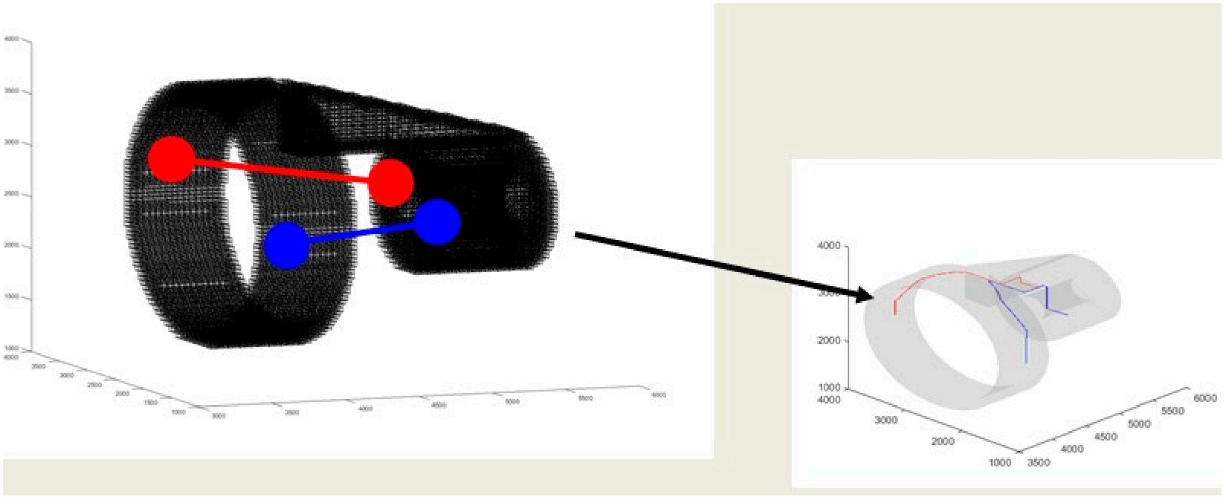


Figure 6: Basic illustration of the creation of 2 connections (indicated in the left picture by red and blue in the large set of black KIZ nodes). Each connection is defined between 2 connectors in the fan- and core-zones, and the resulting optimum routed dressings (also in red and blue in the right picture) are calculated within the KIZ (KIZ is indicated by the grey shaded surfaces in the right picture).

5. Aero-engine equipment, connectors and dressings

The aero-engine equipment considered in this study are of various types, like electric equipment, hydraulic equipment or pneumatic equipment. Some examples are electric components (e.g. electric generators), electronic boxes (e.g. for control units), oil coolers (e.g. for air-oil heat exchangers), or air valves (e.g. for bleed air systems). Typically these equipment are interconnected in different ways. For example an electric generator may be connected by electric harnesses to electronic power consumers and to electronic control units and by oil pipes to a heat exchanger. For these interconnections, or dressings, the equipment has several connectors that are located somewhere on the outer surface of the equipment.

In the design phase of an engine, the locations of these connectors on an equipment may be not fully fixed, but yet to be determined in combination with the placement and orientation of the equipment in the engine installation. Therefore it is important in engine installation studies to have flexibility in the definition of the locations of these connectors. Normally these equipment are designed and defined in CAD software, in which highly detailed definitions of geometry and properties can be created. These details are less important in engine installation studies, in which slightly simplified geometric definitions of geometry and connectors may be sufficient. Nevertheless, for easy adoption of the equipment information as readily available from its CAD model, the automated translation of this CAD model to the equipment's simplified geometric representation, for example by surface triangulation stored in an STL file, is an efficient way for incorporating equipment in engine installation studies. As an example, a basic solid CAD model of an electric generator is given in figure 7, together with its simplified surface triangulated geometric representation and its connectors' locations.

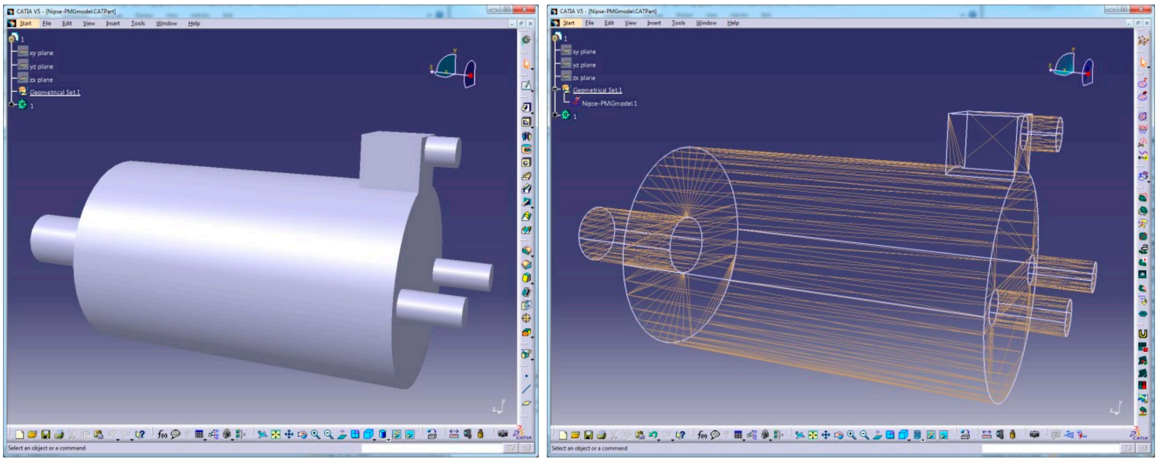


Figure 7: Basic illustration of a solid CAD model (left) of an electric generator. The simplified surface triangulated geometric representation (right) of this electric generator can be automatically generated in common CAD software like Catia [15].Connector locations are included in this CAD model as the small cylindrical parts on the right, representing the electric connector location (on the top-right) and the hydraulic (oil-in and oil-out) connector locations (on the lower-right).

6. Graph-based routing

In engine installation studies, various possible locations of the equipment in the KIZ and of the connectors on the equipment should be easily defined and the feasibility of the required dressings (i.e. interconnections) and their optimal pathways (i.e. routes) should be quickly determined. This can be achieved with the use of the KIZ graph and the equipment' STL files. For example, electric generators and oil cooler heat exchanger equipment can be easily positioned in the 3D KIZ volume and the KIZ graph can be updated correspondingly, i.e. accounting for the space occupation of these equipment and the locations of their connectors. This is illustrated in figure 8.

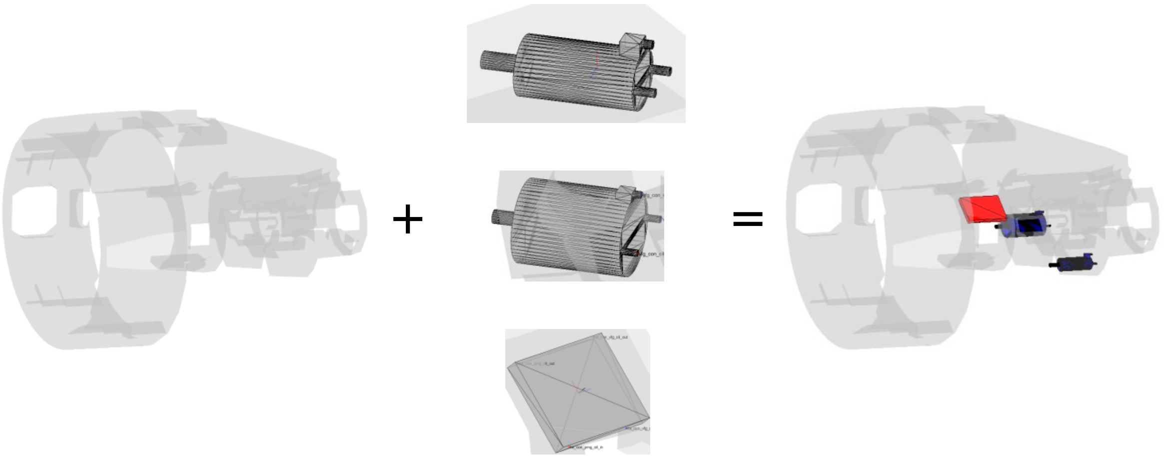


Figure 8: Basic illustration of the 3D KIZ volume (left) and the triangulated geometries of 2 electric generators and 1 oil cooler heat exchanger equipment (middle), and their positioning in the 3D KIZ volume (right).

Appropriate connections between these equipment can then be easily defined, simply by relating the desired connectors to each other, for example for an oil pipe between the oil connectors of the generator and the heat exchanger, as illustrated in figure 9.

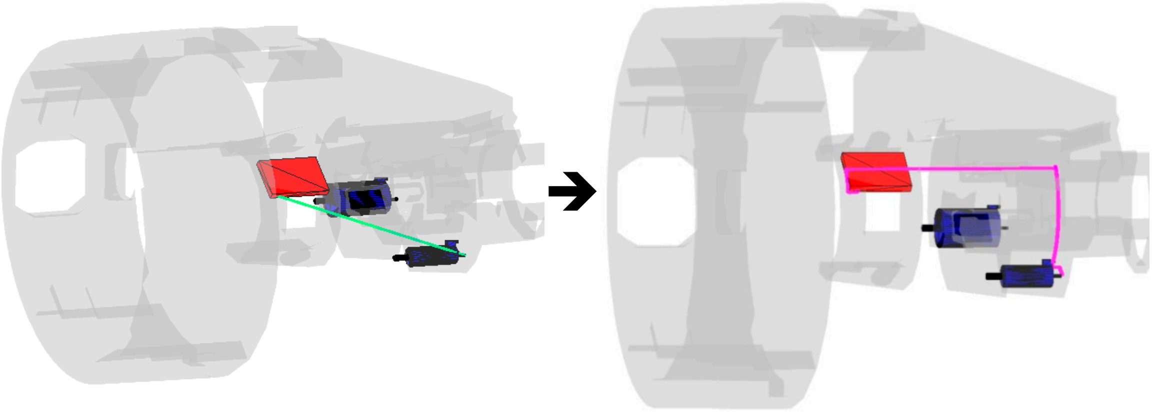
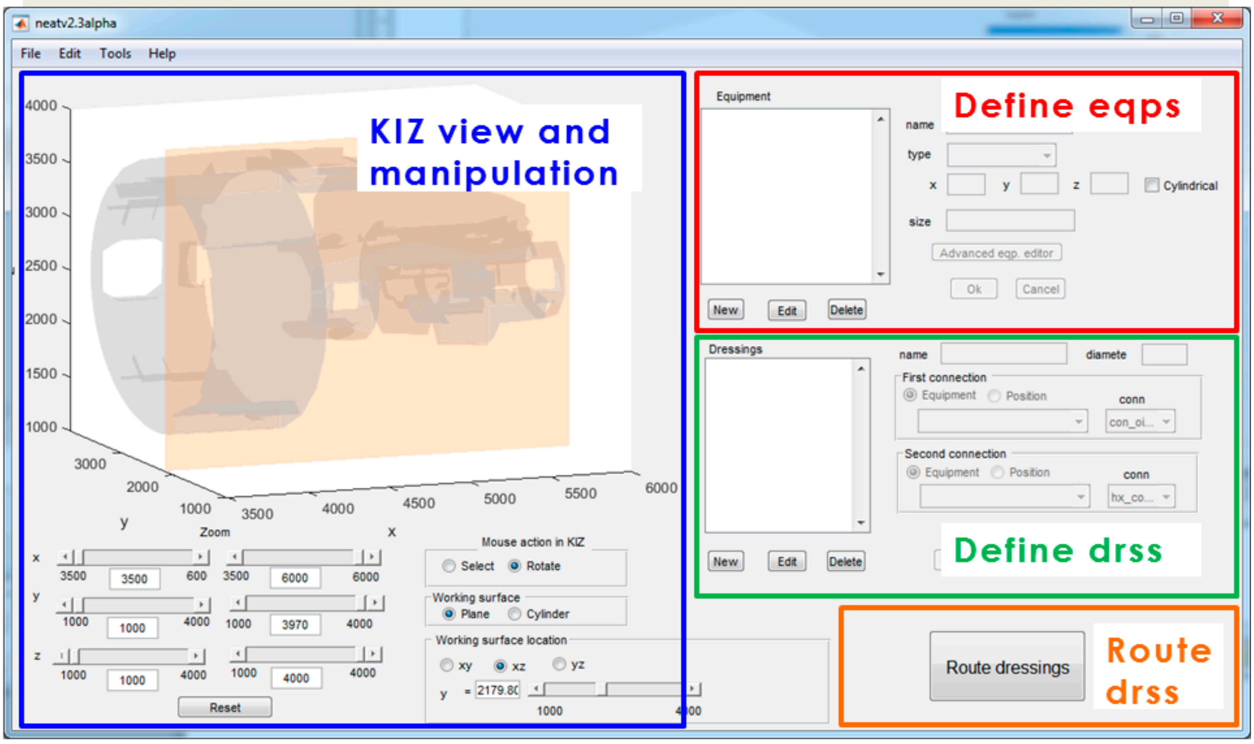


Figure 9: Illustration of the definition of a connection (left picture) for an oil pipe between the oil connectors of the generator (blue cylinder) and the heat exchanger (red box). Also the resulting actual oil pipe of 20mm diameter routed through the KIZ is shown (right picture).

The routing algorithm can then be applied to automatically determine the optimal route for the oil pipe properly through the KIZ and avoiding the occupied space of other equipment or already existing dressings.

7. Software tool development

The optimisation methodology that is built on the graph based approach has been implemented in a software tool. The implementation of this software tool aims for easy user interaction, allowing design engineers to quickly and intuitively make changes to equipment installations taking into account the relevant installation rules. An implementation of this software tool has been developed in Matlab-R2015b [16]. Matlab provides a good basis for efficient data structures and for graph manipulation and optimisation algorithms. Also the development of interactive graphical user interfaces (GUIs), of functionality for 3D results visualisation and of stand-alone (i.e. outside the Matlab environment) executable programs are well supported. The resulting software tool (NEAT - Next generation engine Equipment Allocation Tool; see figure 10) is operational in Microsoft Windows [17] environments (Windows 7, Windows 10) and can be easily ported to other Matlab-supported environments.



Zoom

x: 3500 [3500] 600 | 3500 [6000] 6000

y: 1000 [1000] 4000 | 1000 [3970] 4000

z: 1000 [1000] 4000 | 1000 [4000] 4000

Mouse action in KIZ: ○ Select ● Rotate

Working surface: ● Plane ○ Cylinder

Working surface location: ○ xy ● xz ○ yz

y = 2179.8C (1000 – 4000)

First connection: ● Equipment ○ Position; conn: con_oi...

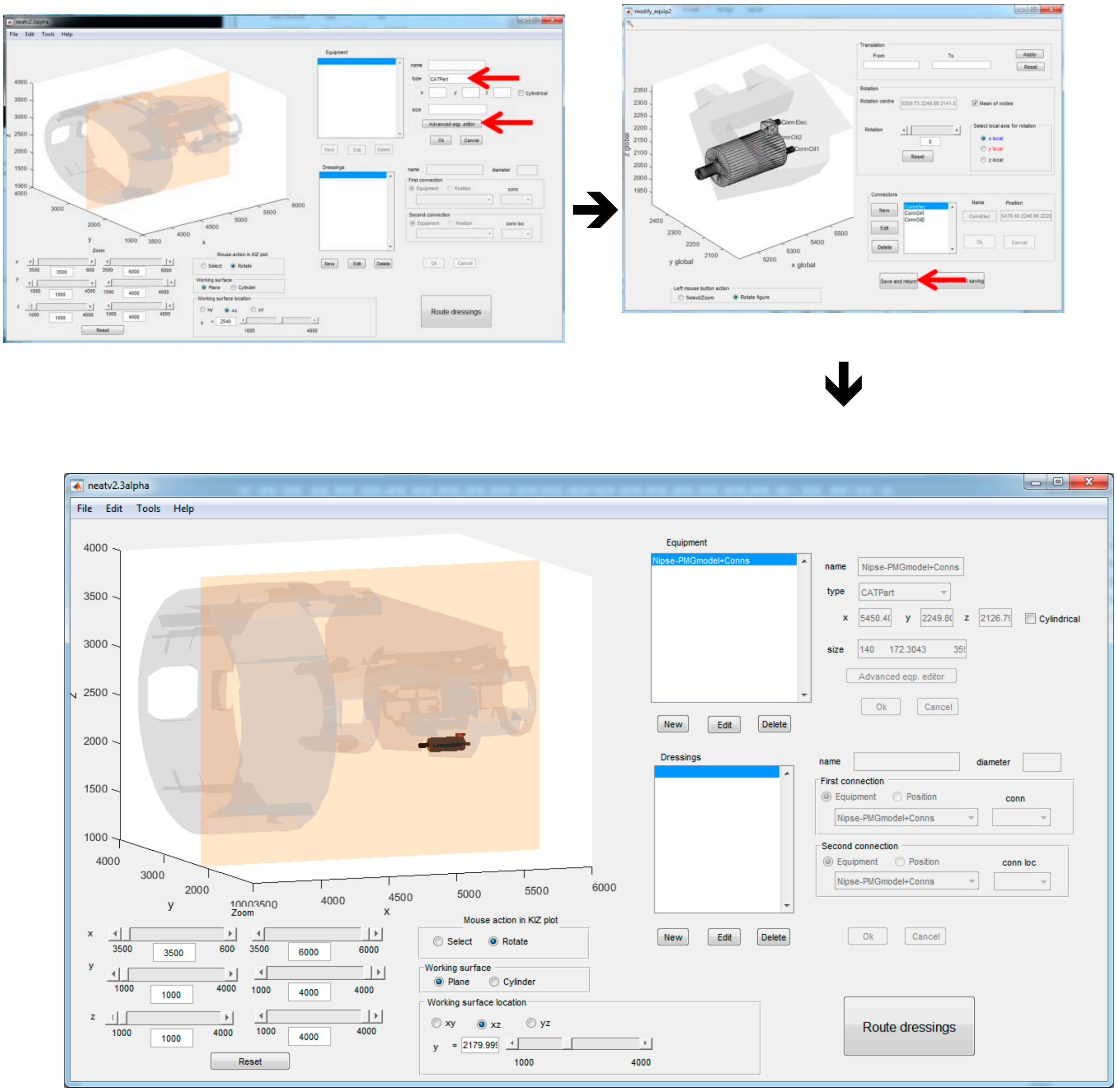
Second connection: ● Equipment ○ Position; conn: hx_co...

Figure 10: Main window of the software tool NEAT. The abbreviations eqps and drss refer to equipment and dressings, respectively.

The NEAT tool provides an interactive visual representation of the KIZ (on the left side in figure 10) that allows for view manipulations like zooming, 3D rotation and point-and-click coordinate selection. The NEAT tool is based on the KIZ of a UHBR engine, but it should be noted that this KIZ can relatively easily be replaced by a KIZ of another engine (or even of a quite different system, for example of an aircraft fuselage or a satellite). NEAT also provides functions (on the right side in figure 10) to define, locate and orientate equipment and to define and route dressings in the KIZ. Configurations can be created and are defined in terms of equipment and connections in the KIZ. These configurations can be saved to and read from Excel files. If the dressings, as defined by the connections and routed by the graph algorithms, have been generated, then the resulting installation can be saved to (or read from) specifically formatted binary data file (based on Matlab's mat-file [16] format). The NEAT tool also provides various other types of settings and manipulations for the positioning and routing of equipment and dressings, such as applying "weights" or "penalty factors" in certain areas of the KIZ graph to define preferred or dis-favoured routing directions. In this way certain accommodation constraints related to the assembly and maintenance of the nacelle can be taken into account in the software. For example the routing of dressings can be prioritized to smaller radial positions resulting in routes that are located more in the central area of the KIZ. Also certain regions in the KIZ can be penalized for routing, resulting in dressings that avoid these regions for example because of difficult accessibility for maintenance. Also environmental variables (typically temperatures) can be assigned to regions in the KIZ, which results in dressings that are located only in regions with an allowable environment. The specification of these regions and "penalty factors" must be made by the user, but the routing algorithm then automatically takes these specifications into account. The placement of the different equipment shall be located in the KIZ by the user of the NEAT tool, either by directly defining the equipment box in the tool or by defining the equipment at its appropriate location in CATIA and then import this equipment in the tool. The complete functionality of the tool cannot be fully explained in this paper, but in the next section the main functionality will be demonstrated.

8. (Application case studies) Basic use case on electric-hydraulic components

We will consider a relatively simple case study, in which the installation of some electric and hydraulic equipment in the KIZ is investigated. First, an electric generator equipment that has been defined in a Catia CAD model, as was previously shown in figure 7, is imported into NEAT and positioned in the desired location and orientation in the core zone part of the KIZ. Also some connectors are defined on the electric generator, which are used for connecting the electric harnesses and the oil pipes for cooling. Illustration of this generator equipment definition in NEAT is given in figure 11.



Equipment: Nipse-PMGmodel+Conns

name: Nipse-PMGmodel+Conns

type: CATPart

x 5450.4(y 2249.8(z 2126.7(

☐ Cylindrical

size: 140 172.3043 35(

First connection: ● Equipment ○ Position — Nipse-PMGmodel+Conns

Second connection: ● Equipment ○ Position — Nipse-PMGmodel+Conns

Mouse action in KIZ plot: ○ Select ● Rotate

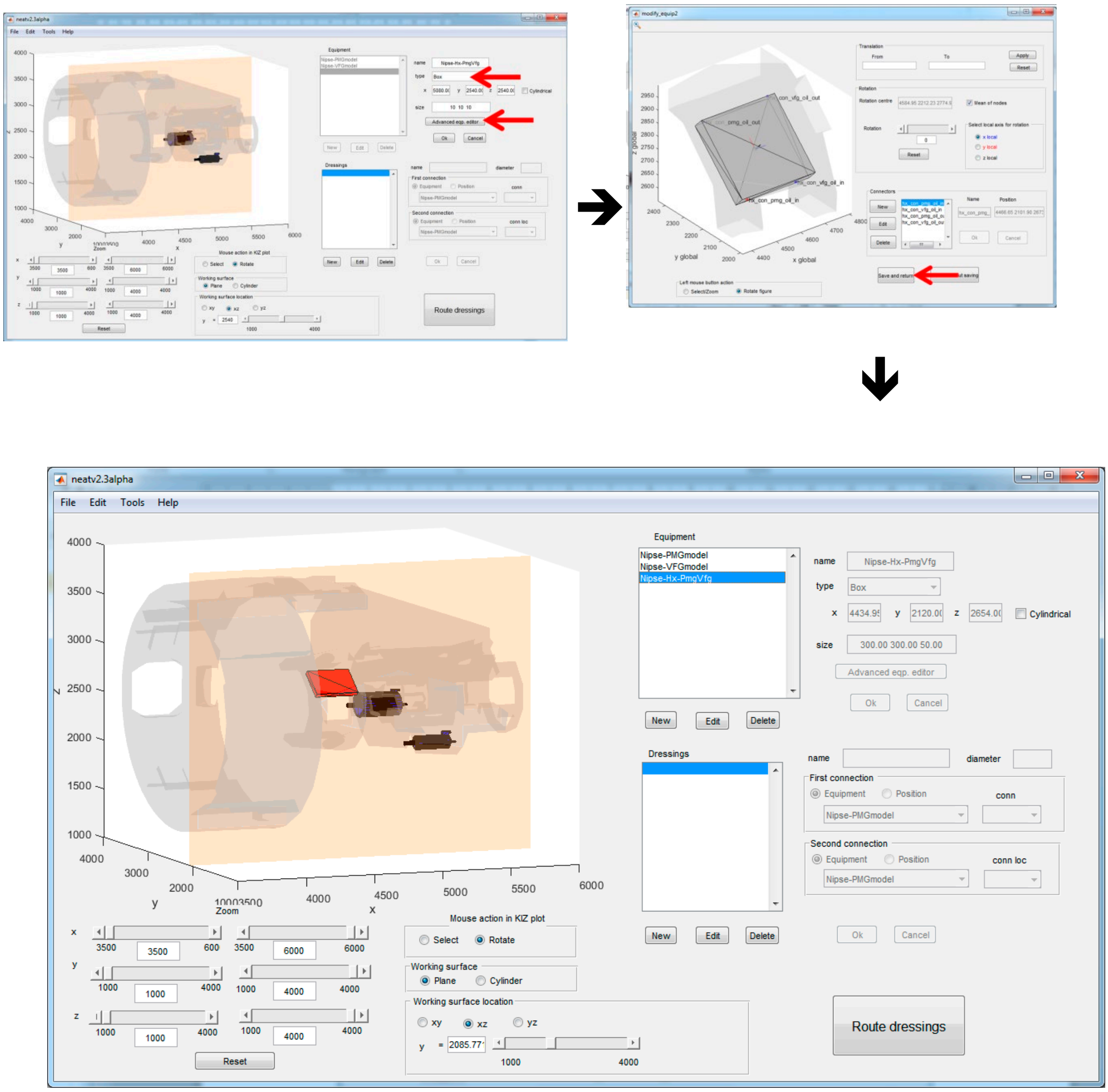
Working surface: ● Plane ○ Cylinder

Working surface location: ○ xy ● xz ○ yz

y = 2179.99(

Figure 11: Illustration of the generator equipment definition in NEAT, where the equipment geometry is imported from a CAD model(upper left picture), moved into the correct location/orientation and devised with electric and oil connectors (upper right picture) and then saved in the KIZ (lower picture).

Subsequently, another slightly different electric generator equipment is imported into NEAT, is devised with electric and oil connectors, and is positioned in the core zone part of the KIZ. Then a heat exchanger equipment is defined in NEAT, as is illustrated in figure 12. This heat exchanger equipment is not imported from a CAD model but directly defined in NEAT as a flat box shape, which is a strongly simplified representation of its approximate geometry. Also 4 oil connectors are defined on the heat exchanger (for connections with both the generators).



Equipment
Nipse-PMGmodel
Nipse-VFGmodel
Nipse-Hx-PmgVfg

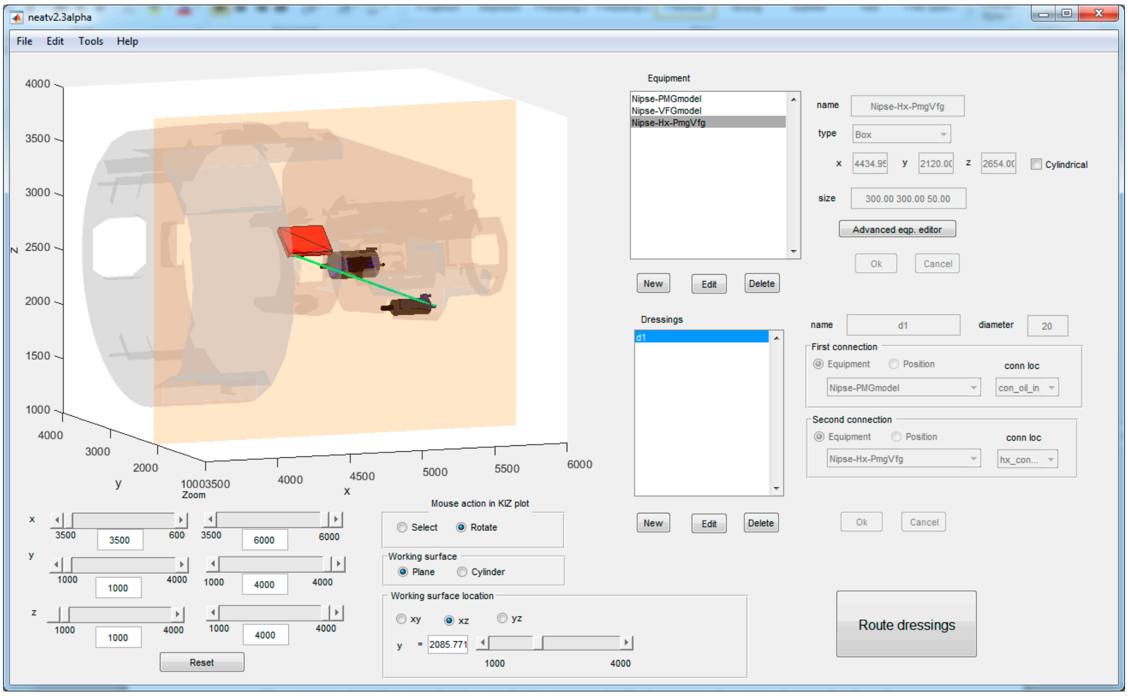
name Nipse-Hx-PmgVfg
type Box
x 4434.95 y 2120.00 z 2654.00 ☐ Cylindrical
size 300.00 300.00 50.00

First connection: ● Equipment ○ Position — Nipse-PMGmodel
Second connection: ● Equipment ○ Position — Nipse-PMGmodel

Mouse action in KIZ plot: ○ Select ● Rotate
Working surface: ● Plane ○ Cylinder
Working surface location: ○ xy ● xz ○ yz — y = 2085.77

Figure 12: Illustration of the heat exchanger equipment definition in NEAT (upper left picture), where the equipment geometry is directly created in NEAT and moved into the correct location/orientation and devised with oil connectors (upper right picture) and then saved in the KIZ (lower picture).

After the equipment have been defined and positioned correctly in the KIZ, dressings can be easily defined by creating connections between equipment connectors. For example an oil pipe dressing between the first generator and the heat exchanger can be defined, as shown in figure 13.



Equipment: Nipse-PMGmodel, Nipse-VFGmodel, Nipse-Hx-PmgVfg

name Nipse-Hx-PmgVfg
type Box
x 4434.95 y 2120.00 z 2654.00 ☐ Cylindrical
size 300.00 300.00 50.00

Dressings: d1

name d1 diameter 20

First connection: ● Equipment ○ Position; Nipse-PMGmodel; conn loc con_oil_in

Second connection: ● Equipment ○ Position; Nipse-Hx-PmgVfg; conn loc hx_con...

x: 3500 [3500] 600; 3500 [6000] 6000
y: 1000 [1000] 4000; 1000 [4000] 4000
z: 1000 [1000] 4000; 1000 [4000] 4000

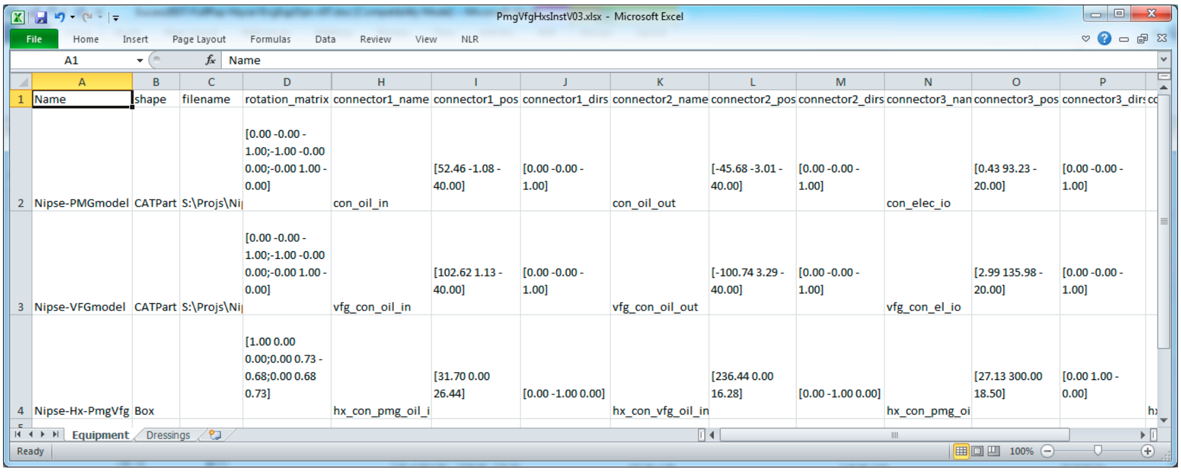
Mouse action in KIZ plot: ○ Select ● Rotate

Working surface: ● Plane ○ Cylinder

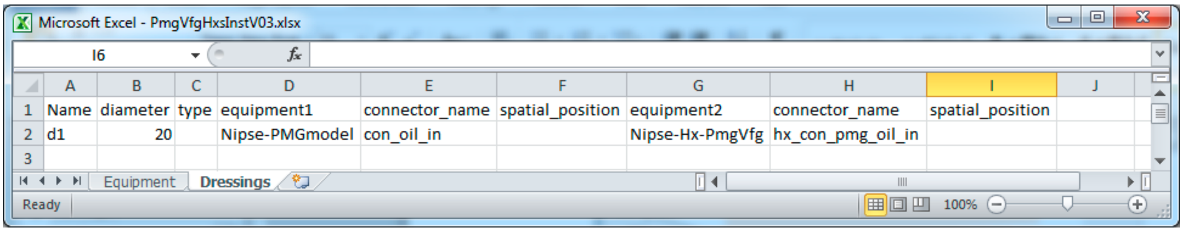
Working surface location: ○ xy ● xz ○ yz; y = 2085.771 (1000 – 4000)

Figure 13: Illustration of the definition of an oil pipe dressing between the first generator and the heat exchanger, which can be easily defined by creating connections between the equipment connectors (indicated by the green line in the 3D view on the upper left).

The resulting configuration data, containing all the relevant data of the equipment and dressings (like their name, shape, filename, position, size, type, connectors, etc.), can be directly exported from NEAT to Excel [18]. The resulting configuration Excel file (as is shown in figure 14) can also be manipulated (i.e. equipment and dressings can be changed or added) and imported back into NEAT. This allows for easy creation, editing and exchange of NEAT configuration models outside the NEAT tool.



Name	shape	filename	rotation_matrix	connector1_name	connector1_pos	connector1_dirs	connector2_name	connector2_pos	connector2_dirs	connector3_nan	connector3_pos	connector3_dirs
Nipse-PMGmodel	CATPart	S:\Projs\Ni	[0.00 -0.00 - 1.00;-1.00 -0.00 0.00;-0.00 1.00 - 0.00]	con_oil_in	[52.46 -1.08 - 40.00]	[0.00 -0.00 - 1.00]	con_oil_out	[-45.68 -3.01 - 40.00]	[0.00 -0.00 - 1.00]	con_elec_io	[0.43 93.23 - 20.00]	[0.00 -0.00 - 1.00]
Nipse-VFGmodel	CATPart	S:\Projs\Ni	[0.00 -0.00 - 1.00;-1.00 -0.00 0.00;-0.00 1.00 - 0.00]	vfg_con_oil_in	[102.62 1.13 - 40.00]	[0.00 -0.00 - 1.00]	vfg_con_oil_out	[-100.74 3.29 - 40.00]	[0.00 -0.00 - 1.00]	vfg_con_el_io	[2.99 135.98 - 20.00]	[0.00 -0.00 - 1.00]
Nipse-Hx-PmgVfg	Box		[1.00 0.00 0.00;0.00 0.73 - 0.68;0.00 0.68 0.73]	hx_con_pmg_oil_i	[31.70 0.00 26.44]	[0.00 -1.00 0.00]	hx_con_vfg_oil_in	[236.44 0.00 16.28]	[0.00 -1.00 0.00]	hx_con_pmg_oi	[27.13 300.00 18.50]	[0.00 1.00 - 0.00]



Name	diameter	type	equipment1	connector_name	spatial_position	equipment2	connector_name	spatial_position
d1	20		Nipse-PMGmodel	con_oil_in		Nipse-Hx-PmgVfg	hx_con_pmg_oil_in	

Figure 14: Illustration of the configuration Excel file containing the definitions of the 2 generator equipment, the heat exchanger and the oil pipe dressing as shown in figure 13.

It should be noted that the configuration Excel file does not contain the information of the routed dressings, but only the dressing definitions (like name, connectors, diameter etc) that are needed by the routing algorithm to determine the routing of a dressing. For example the oil pipe dressing that was defined can be easily routed through the KIZ (by the button “Route dressings”), yielding the routed dressing as was shown above in figure 9 (right picture). Similarly the other dressings between the equipment can be defined and their routes can be determined. Subsequently the routes for the oil dressings for both the generators from and to the heat exchanger are determined, as well as the routes for the electric dressings for both the generators to the so-called junction box (i.e. a surface on the KIZ in the bifurcation zone where the systems interface between the engine and the aircraft is located). The resulting installation of the 3 equipment and the 6 routed dressings is shown in figure 14. For the further processing of these preliminary design results of the installed equipment and the routed dressings, it is necessary to analyse these results in more detail. Therefore the resulting installation can be directly exported to a Catia CAD model (as is also shown in figure 15) allowing an engine OEM to incorporate the installation in their more detailed engine and nacelle design processes.

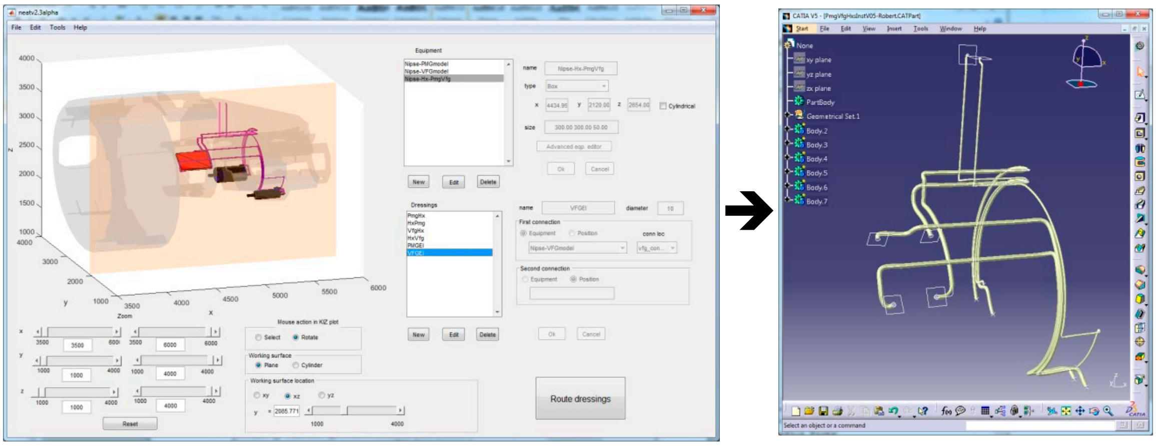


Figure 15: Illustration of the resulting installation of the 3 equipment and the 6 routed dressings in NEAT (left picture) and the routed dressings exported to a Catia CAD model (right picture).

It should be noted that the installed equipment and the routed dressings as shown in this use case are simplified examples, not intended to represent a real engine equipment installation but merely to demonstrate the possibilities for preliminary design and routing of such equipment installations. Installations with higher numbers of more realistic components and with more complex installation rules are being considered in further ongoing investigations.

(8. Results)

The preliminary design process with the NEAT tool that was followed in this use case for the user-interactive installation of equipment and automated routing of dressings could be performed in limited amount of time. The interactive import of the generator equipment from already available CAD parts and their proper positioning in the KIZ, and the interactive creation and positioning of the simplified (box) equipment for the heat exchanger required less than 1 man-hour of work. The automated routing of the 6 dressings required less than 1 minute computation time on a standard PC (Intel Core i5-3320M CPU@2.6GHz, 8GB RAM). Of course these numbers will increase for more complex installations, but it has been demonstrated that the functionality for such preliminary installation design is operational and intuitive such that it can be easily operated with a quite steep learning curve. Also it is expected that reasonable response times of the computational operations are maintained also for larger installations.

9. Advanced use case on NIPSE-ETRAS components

The NEAT tool is also presented in some more detail with a design case from the NIPSE project of the so-called ETRAS (electric thrust reverser actuation system) installation. Both the efficiency and flexibility of the NEAT tool and of the interface with CATIA are demonstrated in this design case. The ETRAS design case that is considered here is a bit more advanced installation example with 8 equipment and simple box and cylinder shaped equipment geometries. It should be noted that also much larger numbers of more complex equipment geometries can be handled in the same way.

When defining the installation design of the ETRAS equipment in the UHBR engine's fan-zone, then typically a CAD system like CATIA is used for this. The ETRAS equipment have to be positioned and interconnected inside the fan-zone of the KIZ. As an example, the ETRAS equipment considered in this design case are shown in Figure 16 and are available as a CATIA assembly. The 8 equipment of the ETRAS are positioned in the UHBR fan-zone and are typically defined and located in the model by the operator in CATIA. The identifiers for the 8 ETRAS equipment are as follows (the exact meaning of these identifiers is not relevant for the present paper):

1. TRCU
2. ETRAS-LOWER LHS ACTUATOR
3. ETRAS-LOWER RHS ACTUATOR
4. ETRAS-PMDU
5. ETRAS-TLS
6. ETRAS-UPPER LHS ACTUATOR
7. ETRAS-UPPER RHS ACTUATOR
8. NECU

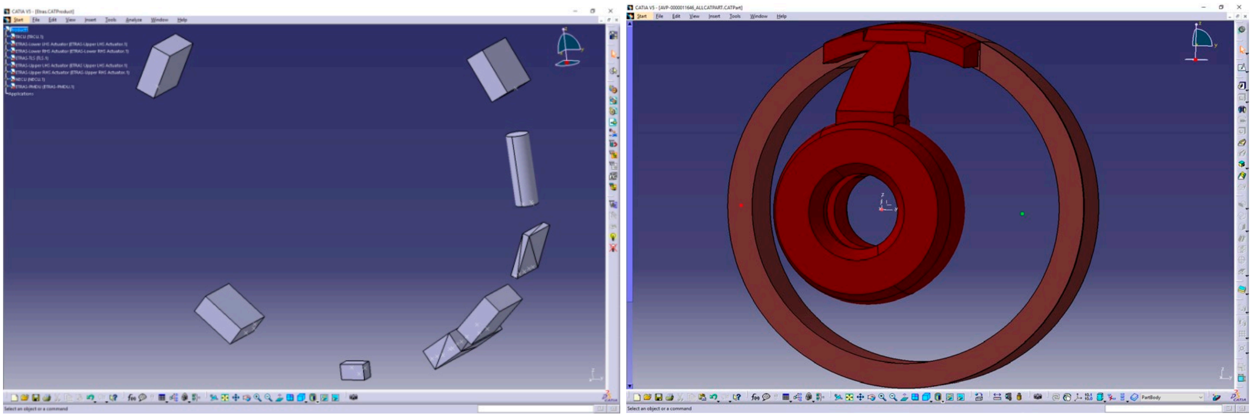
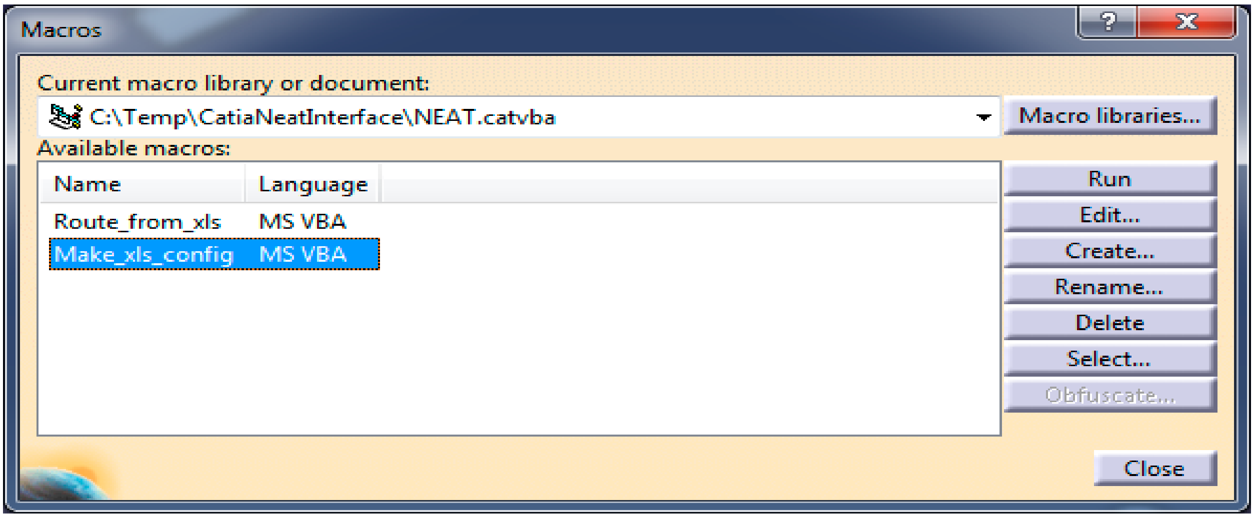


Figure 16: Left picture: illustration of the NIPSE ETRAS-system equipment that are typically available as a CATIA assembly. Right picture: illustration of the KIZ, with the large outer ring being the fan zone in which the ETRAS equipment shall be fitted.

Some key challenges in this ETRAS system design process are:

1. How to easily position (and re-position!) the ETRAS equipment inside the KIZ?
2. How to easily define the dressings (i.e. the interconnections) between the ETRAS equipment?
3. How to easily route (and re-route!) the dressings between the ETRAS equipment inside the KIZ?

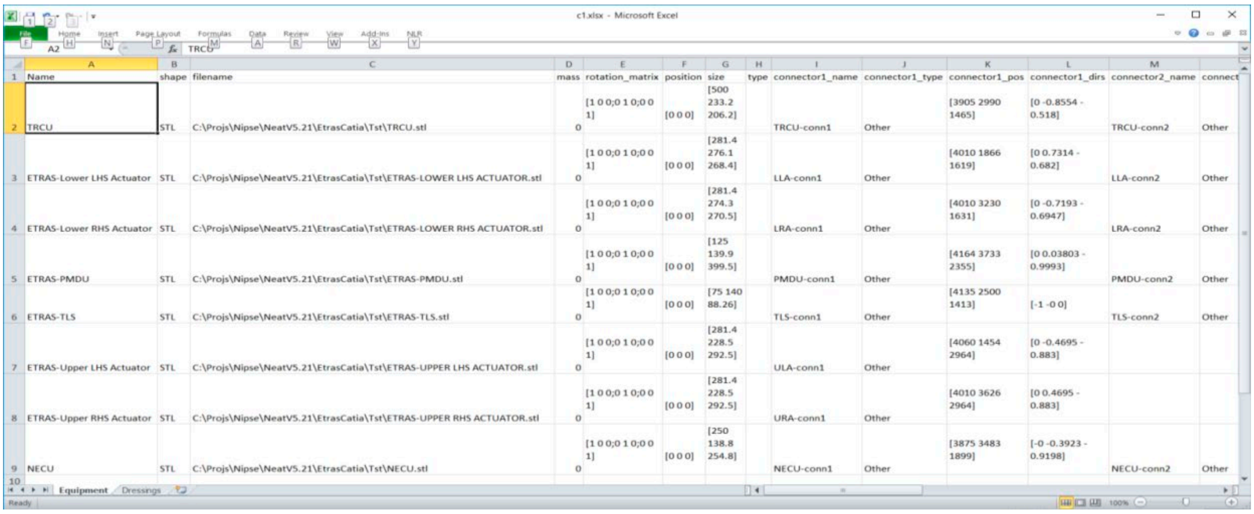
With respect to challenge number 1: the positioning of each of the 8 ETRAS equipment inside the KIZ can be done interactively in CATIA. But the definition and optimal routing of dressings (e.g. harnesses) between these equipment is not necessarily done very well in CATIA. Especially if the equipment still have to be re-positioned and consequently all the dressings need to be re-optimised, i.e. re-routed. Because the automatic routing optimisation of dressings is well supported by the NEAT software tool, it is important to have efficient and flexible interfaces between the CATIA and NEAT software environments. These interfaces allow for the easy exchange of equipment, dressings and assembly information between these tools. Besides the interfaces, also a concise definition of dressings shall be supported: items like the name of the dressing, the equipment and connector from which and to which it connects, diameter and mass shall be identified. Because the ETRAS equipment are available in CATIA, all the equipment's properties (name, size, geometry, position, orientation, connectors) are also available in CATIA and can be automatically exported from CATIA to NEAT. This export is achieved by automatic generation of a configuration Excel file in which the definitions (name, shape, STL filename, etc) of all the equipment are included in the Equipment sheet (see figure 17). This automatic generation is executed in CATIA with VBA scripts (see figure 17) and also creates the STL files (for geometry) and IGES files (for connectors) of all the equipment directly from the CATIA CATpart files. With respect to challenge number 2: In the configuration Excel file the desired dressings can be easily added by inserting rows with the dressing specifications (name of the dressing, names of the equipment and connectors between which the dressing must be created, dressing diameter, etc) in the Dressings sheet (see figure 17). This shall be done manually by the user for each of the dressings that are desired for the installation.



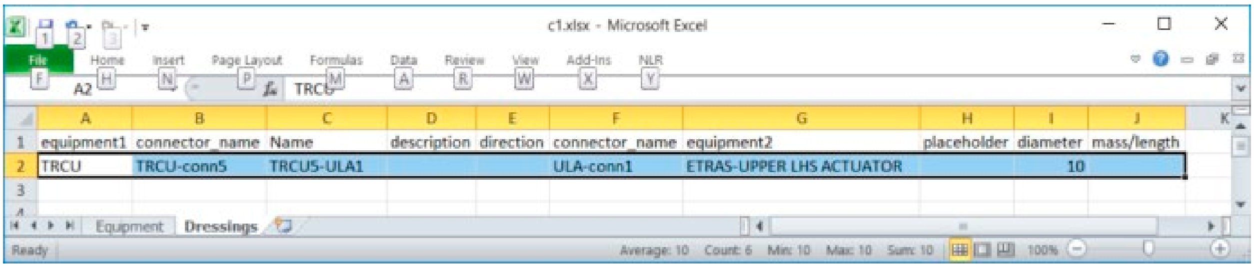
Current macro library or document:
C:\Temp\CatiaNeatInterface\NEAT.catvba

Available macros:

Name	Language
Route_from_xls	MS VBA
Make_xls_config	MS VBA



	A	B	C	D	E	F	G	H	I	J	K	L	M	
1	Name	shape	filename	mass	rotation_matrix	position	size	type	connector1_name	connector1_type	connector1_pos	connector1_dirs	connector2_name	connect
2	TRCU	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\TRCU.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[500 233.2 206.2]		TRCU-conn1	Other	[3905 2990 1465]	[0 -0.8554 -0.518]	TRCU-conn2	Other
3	ETRAS-Lower LHS Actuator	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-LOWER LHS ACTUATOR.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[281.4 276.1 268.4]		LLA-conn1	Other	[4010 1866 1619]	[0 0.7314 -0.682]	LLA-conn2	Other
4	ETRAS-Lower RHS Actuator	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-LOWER RHS ACTUATOR.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[281.4 274.3 270.5]		LRA-conn1	Other	[4010 3230 1631]	[0 -0.7193 -0.6947]	LRA-conn2	Other
5	ETRAS-PMDU	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-PMDU.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[125 139.9 399.5]		PMDU-conn1	Other	[4164 3733 2355]	[0 0.03803 -0.9993]	PMDU-conn2	Other
6	ETRAS-TLS	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-TLS.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[75 140 88.26]		TLS-conn1	Other	[4135 2500 1413]	[-1 -0 0]	TLS-conn2	Other
7	ETRAS-Upper LHS Actuator	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-UPPER LHS ACTUATOR.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[281.4 228.5 292.5]		ULA-conn1	Other	[4060 1454 2964]	[0 -0.4695 -0.883]		
8	ETRAS-Upper RHS Actuator	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\ETRAS-UPPER RHS ACTUATOR.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[281.4 228.5 292.5]		URA-conn1	Other	[4010 3626 2964]	[0 0.4695 -0.883]		
9	NECU	STL	C:\Projs\Nipse\NeatV5.21\EtrasCatia\Tst\NECU.stl	0	[1 0 0;0 1 0;0 0 1]	[0 0 0]	[250 138.8 254.8]		NECU-conn1	Other	[3875 3483 1899]	[-0 -0.3923 -0.9198]	NECU-conn2	Other
10														

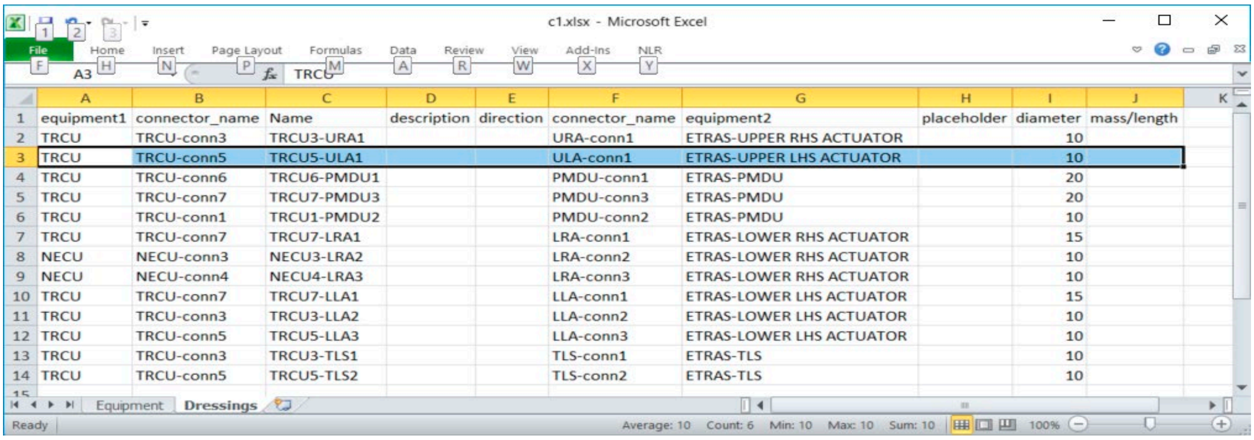


	A	B	C	D	E	F	G	H	I	J	K
1	equipment1	connector_name	Name	description	direction	connector_name	equipment2	placeholder	diameter	mass/length	
2	TRCU	TRCU-conn5	TRCU5-ULA1			ULA-conn1	ETRAS-UPPER LHS ACTUATOR		10		
3											

Average: 10 Count: 6 Min: 10 Max: 10 Sum: 10

Figure 17: Upper picture: Illustration of the VBA script (“Make_xls_config”) in CATIA for the automatic generation of a configuration Excel file with the definitions of all the equipment in the current CATIA assembly. Middle picture: The Equipment sheet in the Excel configuration file contains all the data of the 8 equipment in the ETRAS installation. Lower picture: the Dressings sheet contains all the data of the connections for dressings, in this case only the TRCU5-ULA1 connection of the ETRAS installation is shown.

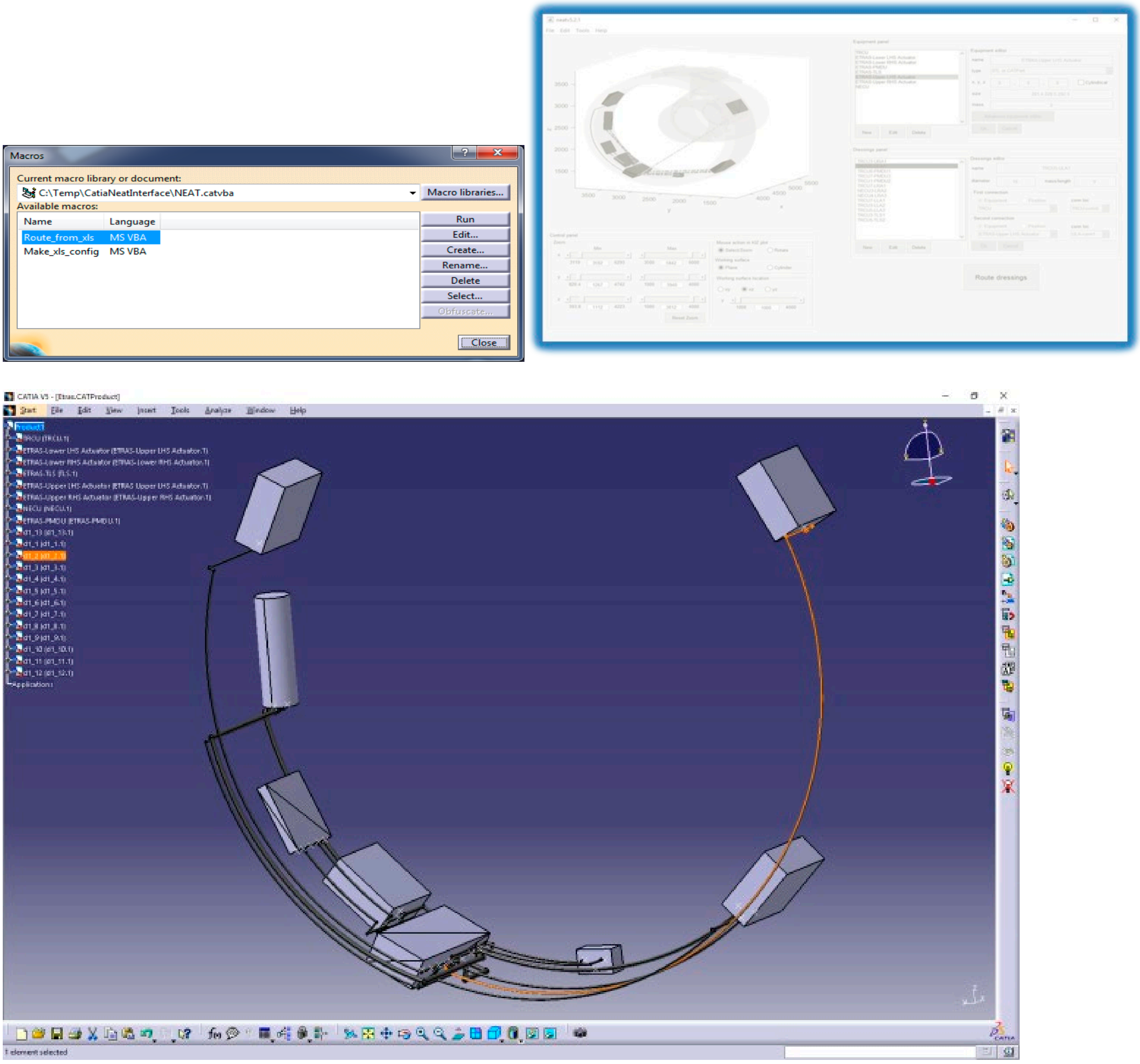
In the Excel configuration file the connections for dressings can be further edited, added or deleted directly in Excel. For example 13 dressings are defined for the ETRAS installation (figure 18).



	A	B	C	D	E	F	G	H	I	J
1	equipment1	connector_name	Name	description	direction	connector_name	equipment2	placeholder	diameter	mass/length
2	TRCU	TRCU-conn3	TRCU3-URA1			URA-conn1	ETRAS-UPPER RHS ACTUATOR		10	
3	TRCU	TRCU-conn5	TRCU5-ULA1			ULA-conn1	ETRAS-UPPER LHS ACTUATOR		10	
4	TRCU	TRCU-conn6	TRCU6-PMDU1			PMDU-conn1	ETRAS-PMDU		20	
5	TRCU	TRCU-conn7	TRCU7-PMDU3			PMDU-conn3	ETRAS-PMDU		20	
6	TRCU	TRCU-conn1	TRCU1-PMDU2			PMDU-conn2	ETRAS-PMDU		10	
7	TRCU	TRCU-conn7	TRCU7-LRA1			LRA-conn1	ETRAS-LOWER RHS ACTUATOR		15	
8	NECU	NECU-conn3	NECU3-LRA2			LRA-conn2	ETRAS-LOWER RHS ACTUATOR		10	
9	NECU	NECU-conn4	NECU4-LRA3			LRA-conn3	ETRAS-LOWER RHS ACTUATOR		10	
10	TRCU	TRCU-conn7	TRCU7-LLA1			LLA-conn1	ETRAS-LOWER LHS ACTUATOR		15	
11	TRCU	TRCU-conn3	TRCU3-LLA2			LLA-conn2	ETRAS-LOWER LHS ACTUATOR		10	
12	TRCU	TRCU-conn5	TRCU5-LLA3			LLA-conn3	ETRAS-LOWER LHS ACTUATOR		10	
13	TRCU	TRCU-conn3	TRCU3-TLS1			TLS-conn1	ETRAS-TLS		10	
14	TRCU	TRCU-conn5	TRCU5-TLS2			TLS-conn2	ETRAS-TLS		10	

Figure 18: Illustration of the updated NEAT Excel configuration file with all the 13 connections in the ETRAS installation.

Once the configuration of the ETRAS installation has been completely defined in the Excel configuration file, the optimized routings for all the dressings can be automatically calculated by the NEAT tool. With respect to challenge number 3: This optimized routing calculation can be directly calculated via a VBA script in the CATIA environment, where the NEAT tool is executed in the background (figure 19). The resulting optimized dressings are also automatically converted to IGES objects and imported into the CATIA assembly.



Current macro library or document: C:\Temp\CatiaNeatInterface\NEAT.catvba

Available macros:

Name	Language
Route_from_xls	MS VBA
Make_xls_config	MS VBA

Figure 19: Upper left picture: Illustration of the optimized routing calculation can be directly calculated via a VBA script in the CATIA environment. Upper right picture: Illustration of the the NEAT tool is executed in the background. Lower picture: Illustration of the resulting optimized dressings that are automatically converted to IGES objects and imported into the CATIA assembly.

The pertinence of the optimal dressings found with the NEAT tool is an important criterion. From analysis of the optimal dressings for this ETRAS case study, it appears that the main constraints that were considered for this case study are correctly taken into account, i.e.:

- All dressings are inside the KIZ
- There is no clash/collision between dressings

- The number of bends in the dressings is minimized
- The total dressing length seems to be minimized.

Although the main constraints for the ETRAS case study were correctly fulfilled, it was found that some of the dressings require some manual corrections by post-processing in CATIA to resolve some local issues like undesired local routes or small clearances between dressings. But thanks to the integration of the NEAT tool with the CATIA environment, the time to prepare the installation of the ETRAS case study on a standard PC (i7-7820HQ CPU @ 2.9GHz) is around five minutes. The calculation time for the routing optimization of all the 13 dressings is around one or two minutes. Finally, an additional minute is needed to create CATpart of the 13 dressings and to import them in the global CATIA assembly. If the same 13 dressings are created manually directly in CATIA on similar hardware, it takes approximately 30min. The time to route these 13 dressings is thus decreased by about 75% (from 30min to 8min).

10. Conclusions and further work

The developments of UHBR engines lead to higher operating temperatures and less space for the installation of engine and nacelle equipment and for maintenance accessibility. An optimisation methodology, built on a graph based approaches and efficient routing algorithms, has been implemented in a software tool (NEAT) that allows for flexible placement of equipment and automated routing of pipes, ducts and harnesses into the restricted volume of UHBR nacelles. A relatively simple case study of the installation of some electric generator and heat exchanger equipment in a representative nacelle KIZ has demonstrated the main functionality and the basic operation of this software tool. Efficient routing optimisation of dressings in the KIZ, requiring in the order of seconds compute time on a standard PC, has been shown. Also another more advanced case study with 8 equipment and simple box and cylinder shaped equipment geometries was considered. In this ETRAS design case the efficiency and flexibility of the NEAT tool and of the interface with CATIA are demonstrated. It was shown for the optimal dressings for the ETRAS case study that the main constraints that were considered in this case study were correctly taken into account. It was also found that with the NEAT tool the optimization of the dressings in the installation of the ETRAS case study required about 8 minutes on a standard PC, which is about 22 minutes (75%) less than the manual creation directly in CATIA of the same 13 dressings. It was concluded from the case studies that the automated routing process allows for quick testing and comparing of many configurations (i.e. modifying the location of equipment), which is something very valuable in the early design phases of an IPPS in order to identify the most optimised configuration in terms of weight, cost, maintenance.

Further developments of the functionality are considered in ongoing investigations, for example to incorporate more advanced constraints for the equipment installation, for example to better control separation between equipment, to better account for temperature restrictions, to better handle bends and supports for dressings etc. Also more realistic installations studies with higher numbers of components and with more complex installation rules are investigated and will be reported in near future publications.

Acknowledgements

The research leading to these results has been done as part of the NIPSE project. This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement N° 636218.

References

[1] NIPSE project, www.nipse.eu.

[2] ENOVAL project, www.enoval.eu.

[3] E.W. Dijkstra, A note on two problems in connexion with graphs, *Numerische Mathematik*, 1: 269–271 (1959).

[4] P.E. Hart, N.J. Nilsson, B. Raphael, A Formal Basis for the Heuristic Determination of Minimum Cost Paths, *IEEE Transactions on Systems Science and Cybernetics SSC4*. 4(2): 100–107 (1968).

[5] N.P. Padhy: *Artificial Intelligence and Intelligent Systems*. Oxford University Press, Oxford (2005).

[6] S. Rudolph, J. Beichter, M. Eheim, S. Hess, M. Motzer and R. Weil, On Multi-Disciplinary Architectural Synthesis and Analysis of Complex Systems with Graph-based Design Languages, in: *Proc. 62. Deutscher Luft- und Raumfahrtkongress (DGLR 2013)*, Stuttgart, September 10-12 (2013).

[7] Z. Zhu, G. La Rocca, M. J. L. van Tooren, A methodology to enable automatic 3D routing of aircraft Electrical Wiring Interconnection System, *CEAS Aeronaut J* (2017).

[8] J. Sieber, European Technology Programs for Eco-efficient Ducted Turbofans, *proceedings from the 22nd International Symposium on Air Breathing Engines*, ISABE 2015, Phoenix, USA. 2015.

[9] C.K. Sain, K. Hoeschler, M. Mischke, Concept Study of Variable Area Fan Nozzle for Ultra-High By-Pass Ratio Turbofan Engine, *proceedings from the 22nd International Symposium on Air Breathing Engines*, ISABE 2015, Phoenix, USA. 2015.

[10] E-BREAK project, https://cordis.europa.eu/project/rcn/103887/factsheet/en

[11] M. Silva, E-BREAK: Engine Breakthrough Components and Subsystems, LEMCOTEC/EBREAK/ENOVAL Common Meeting, 2013.

[12] LEMCOTEC project, *http://www.lemcotec.eu/*

[13] R. v. Bank, S. Donnerhack, M. Carzalens, A. Lundbladh, M. Dietz, LEMCOTEC – Improving the Core-Engine Thermal Efficiency, ASME Turbo Expo 2014, GT2014-25040.

[14] CFM International, https://www.cfmaeroengines.com/engines/cfm56.

[15] Dassault Systèmes, *https://www.3ds.com/products-services/catia*.

[16] Mathworks, *https://nl.mathworks.com/products/matlab.html*.

[17] Microsoft, *https://www.microsoft.com/ windows*.

[18] Microsoft, *https://www.microsoft.com/Excel*.

Dedicated to innovation in aerospace

Royal Netherlands Aerospace Centre

NLR is a leading international research centre for aerospace. Bolstered by its multidisciplinary expertise and unrivalled research facilities, NLR provides innovative and integral solutions for the complex challenges in the aerospace sector.

NLR's activities span the full spectrum of Research Development Test & Evaluation (RDT & E). Given NLR's specialist knowledge and facilities, companies turn to NLR for validation, verification, qualification, simulation and evaluation. NLR thereby bridges the gap between research and practical applications, while working for both government and industry at home and abroad.

NLR stands for practical and innovative solutions, technical expertise and a long-term design vision. This allows NLR's cutting edge technology to find its way into successful aerospace programs of OEMs, including Airbus, Embraer and Pilatus. NLR contributes to (military) programs, such as ESA's IXV re-entry vehicle, the F-35, the Apache helicopter, and European programs, including SESAR and Clean Sky 2.

Founded in 1919, and employing some 600 people, NLR achieved a turnover of 76 million euros in 2017, of which 81% derived from contract research, and the remaining from government funds.

For more information visit: **www.nlr.org**

Postal address
PO Box 90502
1006 BM Amsterdam, The Netherlands
e) info@nlr.nl **i)** www.nlr.org

NLR Amsterdam
Anthony Fokkerweg 2
1059 CM Amsterdam, The Netherlands
p) +31 88 511 3113

NLR Marknesse
Voorsterweg 31
8316 PR Marknesse, The Netherlands
p) +31 88 511 4444

NLR is a registered trade name of Stichting Nationaal Lucht- en Ruimtevaartlaboratorium, Chamber of Commerce No. 41150373. VAT No. NL002760551B01