# MICADO: OVERVIEW OF RECENT DEVELOPMENTS WITHIN THE CONCEPTUAL AIRCRAFT DESIGN AND OPTIMIZATION ENVIRONMENT

F. Schültke*, B. Aigner*, T. Effing*, P. Strathoff*, E.Stumpf*
* RWTH Aachen University, Institute of Aerospace Systems, Wüllnerstr. 7, 52062 Aachen, Germany

## Abstract

This paper presents recent developments within the Multidisciplinary Integrated Conceptual Aircraft Design and Optimization environment (MICADO). MICADO has been developed at the Institute of Aerospace Systems of RWTH Aachen University since 2008; the last full status overview, however, was given in 2012. In the course of the last years, not only the underlying software architecture but also the level of detail in the different disciplines of conceptual aircraft design have been continuously improved. In the field of software architecture, the developer- and user-friendliness have increased. At the same time, improvements in methodology and modules have enabled re-designs of already existing aircraft and evaluation capabilities of different technologies. In addition, the design scope has been extended to smaller aircraft from the CS-23 sector, and the aircraft design analysis competencies have been expanded to economic, ecological, and sociological aspects. An overview of projects in which MICADO has been used shows the manifold applicability of the software.

**Keywords**
Conceptual aircraft design; MICADO

## 1. INTRODUCTION

In 2008, the Institute of Aerospace Systems of RWTH Aachen University initiated the basis for the conceptual aircraft design environment MICADO. Based on minimal user input such as Top-Level Aircraft Requirements (TLARs) and design specifications, MICADO enables the design of consistent CS-25 aircraft on a conceptual design level. Additionally, it is possible to perform fast parameter variations and to optimize existing aircraft designs with regard to various parameters. Besides creating consistent aircraft designs, MICADO is recognized for its holistic technology assessment. These options have been successfully applied in various research and industrial projects.

Although recent PhD theses give overviews of the MICADO framework (e.g. Risse in 2016 [1]), the last publication dedicated exclusively to MICADO was presented in 2012 [2]. Since then, however, MICADO has been continuously improved, which means that these publications no longer reflect the current status. Therefore, the objective of this publication is to give an overview of both changes within the MICADO environment since 2012 and projects in which MICADO has been used.

The recent developments can be divided as follows: In the years from 2012 to 2015, the focus was on methodology improvement and integration of new technologies to the conceptual aircraft design process. Besides, the focus in research projects has also shifted from conceptual design towards technology integration and evaluation. In the last years from 2015 until today, the focus has been mainly on software refactoring and application.

In line with the objective, this publication is structured as follows: First, the status of MICADO from the year 2012 is presented in Ch. 2. Afterward, the developments and changes from 2012 until today are outlined. This part is divided into the general software architecture (cf. Ch. 3) and adaptions within the modules and methodologies (cf. Ch. 4). Finally, Ch. 5 gives an overview of projects in which MICADO has been applied. The publication concludes with an outlook (cf. Ch. 6) to planned developments for the next few years.

## 2. STATUS 2012

A computer-based design environment like MICADO always consists of a certain software architecture and the implemented design philosophy. In order to point out the recent developments of both architecture and methodology in Ch. 3 and Ch. 4, respectively, this chapter initially provides a brief overview of the status of MICADO presented in 2012 [2].

**Software architecture:** The central aspect characterizing the software architecture of MICADO is the stand-alone capability and independence of every tool to ensure modularity and flexibility. For consistent data handling, the input and output data of every tool is stored in one central XML[1] aircraft exchange file (AiX-file), which therefore serves as both a generic description of the aircraft and a central data repository. The user controls the individual modules via their respective XML configuration files, which include not only common control but also specific program settings. The resulting principle of control (vertical arrows) and data (horizontal arrows) flow is schematically depicted in Fig.1.

---

[1] Extensible Markup Language

doi: 10.25967/530093

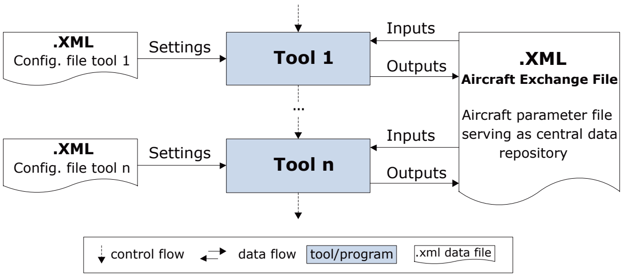


**FIG 1. Principle of control and data flow in MICADO [2]**

This principle enables
a) the exclusion of any tool as long as the required set of input data for the other tools is provided, and
b) the replacement of any tool as long as the responsibility for the specified set of output data is maintained.

Moving on to the code level, all in-house developed tools are implemented in C++. To handle repeated tasks or large datasets and to facilitate programming, in-house developed static libraries are compiled in every tool next to shared XML parser libraries. As long as the XML standards for control settings and the input and output data handling are followed, however, software modules can be implemented in any other programming language with their own libraries. Additionally, external software, such as LIFTING_LINE [3], can be integrated with appropriate wrappers. The described software architecture ensures consistent data management and reasonable computation time.

**Overall aircraft design methodology:** The methodology introduced in 2012 enables clean sheet aircraft designs in a fully automated aircraft design synthesis. This allows for consistent design of a complete aircraft with a minimum of user input or, more precisely, with a minimum set of TLARs as well as design specifications. An overview of the key process chain of MICADO is given in Fig. 2.

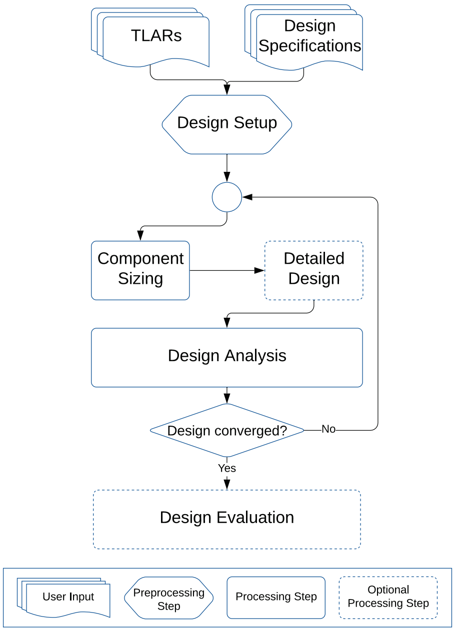


**FIG 2. Overall MICADO process chain**

As can be seen at the very top of Fig. 2, every aircraft design starts by deriving the TLARs to fulfill a specific transport task. In addition, the user specifies qualitative parameters that describe the aircraft concept to be designed; naturally, these should be in line with the previously defined TLARs. The following design setup includes not only the identification of both the design point (wing-loading and thrust-to-weight ratio) and an initial maximum take-off mass (MTOM) but also the design of the fuselage layout. Except for the MTOM, these parameters are kept constant throughout the design process and therefore characterize the selected aircraft concept. The subsequent iterative process includes component sizing programs, optional detailed design modules, and analysis programs; the latter estimate the aerodynamic, mass, and performance characteristics of the previously sized aircraft. The overall aircraft design process is then re-executed until the residua of the convergence parameters MTOM, operating mass empty, fuel, and center of gravity are below a user-defined threshold. Optionally, the resulting converged aircraft can be evaluated in terms of fuel efficiency, costs, or emissions. Also, using a multi-dimensional set of design variables as an input, it is possible to optimize an aircraft design towards user-specified objectives. For this kind of optimization, the so-called NOMAD algorithm (Nonlinear optimization with the MADS algorithm) is implemented as a black-box optimizer [4].

## 3. IMPROVEMENT OF THE SOFTWARE FRAMEWORK

Due to the high quality of both software architecture and overall aircraft design methodology, as described in the previous chapter, we want to emphasize that the fundamental ideas of MICADO remain unchanged. Nevertheless, there has been a potential for improvements. The group members of both MICADO developers and users constantly change, and familiarization and development time is limited. Therefore, the improvements mainly focused on increasing developer- and user-friendliness.

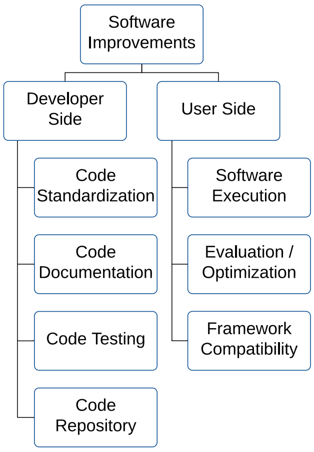


**FIG 3. Overview of the MICADO software improvements**

Figure 3 gives an overview of the improvements introduced to the software framework for developers and users since 2012.

### 3.1. Improvements for developers

From the beginning of MICADO development, the software and with this the number of code lines have continuously grown. In order to still make modifications to the software quickly, easily, reliably, and traceably we initiated improvements in terms of code standardization, documentation, testing, and versioning in 2015.

**Code standardization:** Clean coding [5] is one of the most important fundamentals for keeping such large software projects as MICADO alive. The first concepts for following the rules of clean coding already existed in 2012 by using standardized class structures and libraries. These class and library structures have been extended, and rules have been established that restrict the tasks of every single class and function, taking a further step towards clean code. An overview of the current class structure is given in Fig. 4.

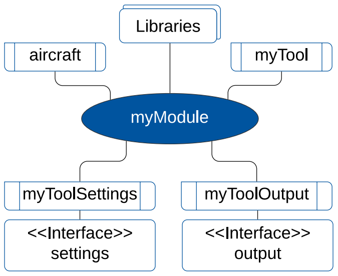


**FIG 4. Overview of the current class structure of a MICADO module**

In contrast to the status of 2012, the classes `settings` and `output` were renamed to `myToolSettings` and `myToolOutput`. These classes contain only tool-specific tasks for settings and output as the general tasks were shifted to respective interfaces using the structure of abstract classes in C++. The `aircraft`-class is now only used to provide the tool-specific aircraft data from the AiX-file to the other classes. All calculation routines were shifted from the `aircraft`-class to the new `myTool`-class. The class `myTool` has two public functions, which are *init* to initialize all class members and *run* to start the calculation process. There can be optional additional classes to keep each class as small as possible and to separate tasks. The entry into each module (`main.cpp`) is standardized as well. First, objects for the classes of `myToolSettings`, `aircraft`, and `myTool` as well as for logging are created, before the tool-object is initialized *(init)* and executed *(run)*. Finally, the object for `myToolOutput` is created, initialized, and the outputs are written.

As can be seen in Fig. 4, additional classes and functions can be addressed through libraries. This was already possible in 2012. At that time, MICADO started with libraries for *aircraft geometry*, *the atmospheric model*, *access to engine data*, and *access to aerodynamic polar data*. These libraries have remained the same in their basic idea and structure. There were only a few smaller adaptions to structural changes outside the libraries. However, the list of libraries was extended. There are additional libraries now for *the communication with the AiX-file*, *the connection to the DLR tool* LIFTING_LINE, *the program execution logging*, *a connection to a database*, *SVG plotting*, as well as for mathematical operations with *the Standard Vector Library and TNT*, and a library for a *unit conversion standard*. The objective of the MICADO team is to extract as many code lines as possible from the single modules. With this

a) the size of the modules is reduced in terms of classes and code lines, which makes the code easier to understand,
b) the development of code can be reduced, and
c) it is more comfortable and less error-prone to make changes throughout the MICADO software.

The last point is the reason for switching from static libraries to dynamic ones. This step has restricted independence of modules a little bit more (now modules need additional files for execution), but for changes in code, modules need not be touched at all. For the future, it is planned to extract more and more lines of code to multiple libraries; this enables to connect not only the modules but also the used methods and functions arbitrarily and to standardize them. Within the different classes, the ILR code style is used for programming. This code style is based on the Google Code Style extended by MICADO-specific rules. The code style is automatically checked using the external programs `cppcheck` and `cpplint`. Furthermore, all modules and their outputs are consistently programmed in English.

**Code documentation:** Proper documentation of the code should be a prerequisite, especially with many people developing the same code. In the first few years of MICADO development, the focus was on fast code generation while neglecting the documentation; this resulted in, e.g., the same aircraft parameters being named differently in the code, making it not only difficult to understand across the modules but also hard to verify. Structures for code documentation were also created, but they were only partially used; even less attention was paid to their compliance and application.

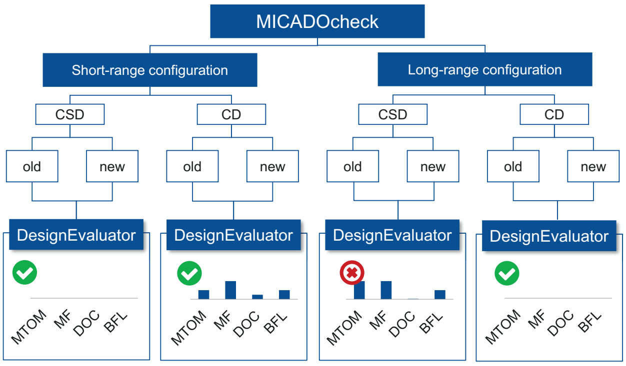
In 2015, new rules were established how to document the code. These rules were based on the structures envisaged at the time, e.g. to use `Doxygen`[2] for automatic documentation generation from code. Each class, each member function, and each member variable has to be described. Furthermore, the task of the module and different execution modes are documented on its main page. The equations that are used within the functions must be documented with reference to the literature from which they are derived. This facilitates reviewing and understanding the code.

---

[2] https://www.doxygen.nl/index.html

**Code testing:** Although the software project was hosted on an SVN repository for versioning and revision control, there were no real control instances and processes for code review and testing. If this state had not been changed, the code would not be manageable within the next few years, and there would be no guarantee for clean, consistent code and validity of the produced data anymore. Every programmer is aware that code testing is a very complex and time-consuming, however, essential procedure. In MICADO, there are three possible levels for code testing. The first one is the overall MICADO software environment with all its modules. The second one is testing every single module itself and, finally, the third one is testing every single function within a module.

A new module called *MICADOcheck* has been implemented to address the first level of code testing. Before new or modified code is published within MICADO, the MICADOcheck module has to be executed. In general, the module designs an aircraft configuration with the already published MICADO version and afterward the same aircraft design with the new or modified MICADO version. Both designs are compared in the end and a report is generated. The programmer has to interpret the results of the report. If the result is as intended, the code can be published. Since the used equations can differ with the desired aircraft configuration, the aircraft mass, or the required aircraft mission range, MICADOcheck can design a short-range and a long-range aircraft. In addition, there is a difference in the applied design logic depending on clean sheet aircraft design (CSD) or re-design of calibrated aircraft (CD). This allows for currently four aircraft designs, which are compared by a `Design Evaluator` in the end. MICADOcheck runs fully automated after the user has copied the modified modules into the project environment. Furthermore, the calculation of the four aircraft designs can be executed in parallel on different CPU-cores to reduce the execution time of MICADOcheck. The full procedure is shown in Fig. 5.



CSD: Clean Sheet Design; CD: Calibrated Design

**FIG 5. Overview of the MICADOcheck module**

In the end, the programmer has to go through the reports and decide if the deviations are reasonable or even intended (cf. green checkmark in Fig. 5) or not (cf. red cross in Fig. 5). If there are unaccountable deviations, the programmer has to revise the code and rerun MICADOcheck. For future release versions of MICADOcheck, it is intended that the program can compare the results calculated with the modified MICADO version also to already existing reference aircraft configurations.

Depending on the execution mode of MICADO (CSD or CD) different calculation paths through the modules are addressed. Thereby, MICADOcheck already implicitly performs a functional test, which is the first step for level-two testing. It is planned to extend the second level and to implement the third level of testing in future release versions of MICADO.

**Code repository:** For code versioning and control, the code of all MICADO modules and libraries was hosted on an own server using an SVN repository. In 2019, the central version control system SVN was switched to the decentralized system Git. Git is now mainly used because the MICADO development group has grown, making it easier to work with Git's branching system simultaneously. There is one repository each for modules, libraries, and engine data. Each repository is set up in a typical approach, having a release branch and a developer branch. This classification is used to release version packages of MICADO modules, libraries, and engine data. In August 2020, version 1.0 of MICADO was released. From this date, large parts of the source code have been published for the "Luftfahrtforschung (LuFo)"-project *UNICADO* as well [6]. As the group of code developers grows, code review becomes increasingly critical. With the use of a version control system, the basis for a code review process is laid. During the *UNICADO* project, it is planned to install a code review process.

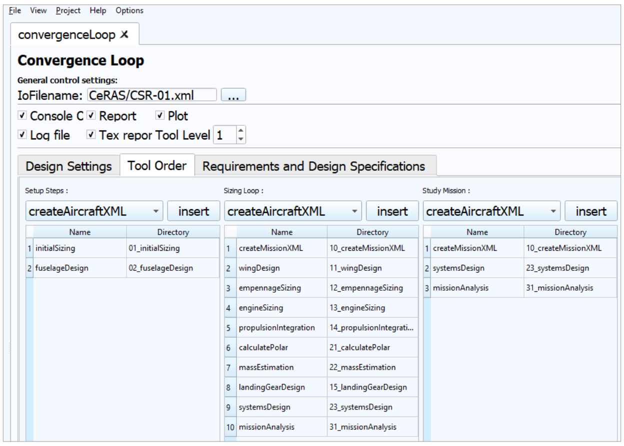
### 3.2. Improvements for users

Besides improving MICADO on the developing side, the software should obviously be as user-friendly as possible, especially for inexperienced users, to support them in obtaining their desired design study results. This has become increasingly relevant in recent years with a growing number of people working with MICADO. Therefore, several improvements have been implemented for executing the software, evaluating and optimizing aircraft designs, and enabling compatibility with other aircraft design frameworks.

**Software execution:** Due to its modular and object-oriented structure, MICADO is a very powerful tool to perform aircraft designs, parameter studies, and optimizations. However, in the past, using MICADO was almost impossible for beginners without the help of an expert user. Executing MICADO in the desired setup, for instance, required adjusting the configurations of the different modules manually in the separate XML configuration files. Additionally, in its default setup, MICADO expects a specific folder structure that had to be built-up by hand for each design study. The software modules were executed via command line or by double-clicking on the individual executable files and were only compatible with Microsoft Windows systems. In the case of calculating a fully converged aircraft design, this meant executing the convergence loop module, which in turn executed the sub-modules in batch mode. This rather cumbersome and unintuitive way of executing MICADO was already recognized in the early days of the framework and addressed with the idea of a graphical user interface (GUI) [2], which, however, was never released.

In 2015, the MICADO development group started establishing a new graphical user interface using the open-source GUI toolkit Qt[3]. The main idea is to support inexperienced users with executing design studies, while at the same time allowing for individual and more advanced configuration of the framework if desired. Within the GUI, the user works on one or more specific projects, each being an aircraft design study with a certain configuration of the underlying tool chain. The GUI combines three essential elements:

- MICADO modules included in the current project: The configuration of the individual program modules can be performed directly in the GUI without having to adjust the XML configuration files manually (see Fig. 6).



Convergence Loop

General control settings:

IoFilename: CeRAS/CSR-01.xml

☑ Console C ☑ Report ☑ Plot

☑ Log file ☑ Tex repor Tool Level 1

Setup Steps : createAircraftXML

| | Name | Directory |
|---|---|---|
| 1 | initialSizing | 01_initialSizing |
| 2 | fuselageDesign | 02_fuselageDesign |

Sizing Loop : createAircraftXML

| | Name | Directory |
|---|---|---|
| 1 | createMissionXML | 10_createMissionXML |
| 2 | wingDesign | 11_wingDesign |
| 3 | empennageSizing | 12_empennageSizing |
| 4 | engineSizing | 13_engineSizing |
| 5 | propulsionIntegration | 14_propulsionIntegrati... |
| 6 | calculatePolar | 21_calculatePolar |
| 7 | massEstimation | 22_massEstimation |
| 8 | landingGearDesign | 15_landingGearDesign |
| 9 | systemsDesign | 23_systemsDesign |
| 10 | missionAnalysis | 31_missionAnalysis |

Study Mission : createAircraftXML

| | Name | Directory |
|---|---|---|
| 1 | createMissionXML | 10_createMissionXML |
| 2 | systemsDesign | 23_systemsDesign |
| 3 | missionAnalysis | 31_missionAnalysis |

**FIG 6. Screenshot of the convergence loop configuration in the MICADO GUI**

- Aircraft data file on which the study is performed: The aircraft dataset at hand can be inspected by opening the respective AiX-file. Additionally, a 3D-view of the aircraft geometry is available.
- Execution results: Once a tool has been executed, the resulting HTML report files can be directly accessed from the GUI. Furthermore, the user has access to the tool log files, giving information about the tool execution, such as warnings or errors.

A second and recent improvement with respect to software execution is the compatibility with Unix systems, opening up MICADO to a larger group of users. This additionally enables the user to run large and computation-intensive studies on compute clusters which only allow Unix workflows.

**Results evaluation and optimization:** With respect to results interpretation and design optimization, MICADO has been widely extended over the past years. Each of the individual MICADO modules generates output plots and short reports containing the main results of their analysis. This facilitates detecting errors, shortcomings, or inconsistencies in the aircraft designs. A summarizing PDF report combines all tool reports in one overview document, which contains the key specifics of an aircraft design. This report can be handed out to project partners, customers, or other externals as a short overview of the aircraft design.

For aircraft design studies and optimizations, the already existing parameter studies on equidistant meshes have been extended by Latin-Hypercube sampling to scan the design space more efficiently. In addition, surrogate modeling using the Kriging process within MICADO was investigated. The work enables the creation of surrogate models for run-time-intensive calculations and software modules to accelerate the design process.

**Compatibility to other frameworks**: Up to this point, MICADO has been described as a flexible and modular but stand-alone framework for aircraft design. However, in various research projects in which the ILR has been involved, only parts of the MICADO analysis were used. Combining MICADO modules with design and analysis competencies from multiple external partners arises challenges, which are described in the following section.

One prerequisite when using different software within the scope of a multidisciplinary project is finding a feasible concept to transfer data among the consortium to ensure consistency and validity of the data provided. The German Aerospace Center (DLR) has been developing the XML-based data storage and exchange format CPACS[4] to address this issue [7]. CPACS has been established as a standard for aircraft data parametrization and is used by entities dealing with aircraft design all over the world. Therefore, an interface between the MICADO AiX-file and CPACS has been created. Within this interface, the CPACS-native libraries TiXi [8] (for parsing the CPACS XML) and TiGL [9, 10] (for access to the CPACS geometry parameters) are used to convert AiX-file data to CPACS and vice versa.

While CPACS is tailored to address the problem of how to exchange data, a second issue when working in a multidisciplinary and multi-partner environment is how to connect the different competencies provided on the process level, i.e. how the different competencies/analyses/tools are combined to obtain the desired results. The Remote Component Environment (RCE) developed at the DLR addresses this challenge [11]. It allows for integration of heterogeneous software modules from different programming languages combining them in a single platform. It is even possible to execute processes across differ-

---

[3] https://qt.io

[4] Common Parametric Aircraft Configuration Scheme

ent organizations without violating any IT security regulations. In the European projects *AGILE*[5] and *AGILE 4.0*[6], some MICADO modules have been successfully integrated with RCE within a larger aircraft design tool chain. In a recent internal project at ILR the entire MICADO convergence loop–which usually is a C++ program module executing the other sub-modules in batch mode–has been substituted by an RCE workflow, facilitating the connection of MICADO with design and analysis software from other organizations in the future.

## 3.3. SUMMARY

The enhancements noted above ensure that MICADO creates verified, validated, and consistent datasets of aircraft designs very quickly. Rules for clean coding were introduced and already applied, improving the software development and maintenance process. In addition, the clean code facilitates entry into module programming for new MICADO group members; the GUI helps them to use MICADO. Finally, automatic report generation and having data and software interfaces simplifies collaboration with external partners.

## 4. IMPROVEMENT OF THE MICADO MODULES

This section describes the recent changes and improvements within the MICADO modules. The structure of this chapter is based on the MICADO flow chart introduced in Ch. 2. Based on Fig 2, a more detailed overview of the MICADO process chain is shown in Fig 7. Within the stated sub-chapters, the recent developments are summarized for each tool in a single paragraph.

### 4.1. Design Setup

Before starting the MICADO loop, the AiX-file has to be created and the design parameters wing-loading and thrust-to-weight ratio have to be set. Changes in these design setup steps are described below.

**AiX-file:** In 2012, aircraft data was stored in an internal database and subsequently transferred to the AiX-file. The connection to this database has been removed and the information has been shifted to multiple AiX-files decreasing the time for data processing. Moreover, a new dynamic adjustment of the AiX-file structure during MICADO runtime allows using the file in a more flexible way.

**Initial sizing:** Physics-based methods for aerodynamics, engine-, and mission performance were integrated into the initial sizing process to more accurately calculate the take-off mass for a given design point. In addition, the choice of the final aircraft design point has been enhanced by a deterministic approach, searching for a minimum take-off mass as an objective function while the boundaries due to the requirements are applied. Finally, a probabilistic method has been included to be able to consider the uncertainties when estimating the boundary constraints from the set of requirements.

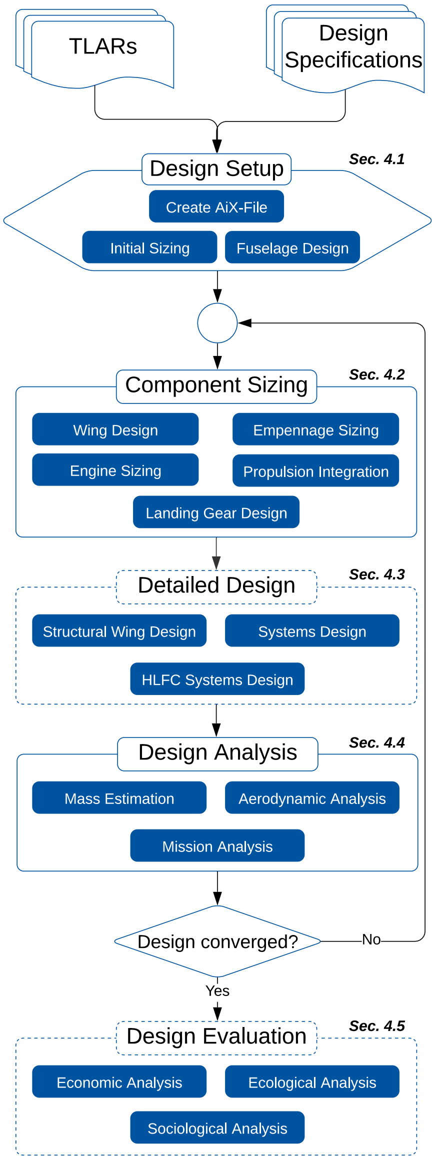


FIG 7. MICADO process chain with tools

**Fuselage design:** Since the center of gravity of the fuselage has a large effect on overall aircraft performance, the fuselage interior design has been extended by using component masses of seats, galleys, and lavatories to calculate a center of gravity for each single passenger class. Furthermore, the interior design has been improved by giving the user a choice between different types of seats, lavatories, galleys, and exits. The cargo deck has been divided into several cargo compartments to be able to use different container types.

Besides interior design, the design of the outer fuselage geometry shape has been adapted by introducing a K-factor that determines the rounding of the corners of a rectangle. This allows for a smooth design transition from a rectangular to a round/elliptical cross-section.

---

[5] http://agile-project.eu

[6] http://agile4.eu

### 4.2. Component sizing

Within the component sizing block of the MICADO environment, the modules for the design and the sizing of the main wing, the horizontal and vertical tailplane, the landing gear, as well as the module for integration of the propulsion system have been extended in recent years.

**Wing design:** For studies on modifications to an existing wing geometry, the wing design and sizing module has been supplemented with corresponding functionalities. In addition to generic initial wing design, the user can now decide whether to scale the whole wing area or only the wingspan for aircraft with predefined wing geometries. The re-sizing mode includes options, for example, to adjust the wing area either to a constant wing loading or to scale it by a user-defined scaling factor. Moreover, the geometric scaling center, as well as options on re-positioning the wing with respect to the aircraft's center of gravity, can be chosen. Concerning span scaling, the user can define a particular span, scale the existing span by a factor, or scale the span by prescribing an aspect ratio. The span scaling mode further allows the user to choose between a constant taper ratio or tip chord length.

**Empennage sizing:** Similar to the wing design module, the empennage sizing module, which designs and sizes both horizontal and vertical tailplane, was supplemented with a re-sizing mode for a given geometry of the aircraft tail. The empennage re-sizing mode, however, has a less extensive range of features. The re-sizing for both horizontal and vertical tailplane is carried out under the assumption of a constant volume coefficient for each part of the tail. The user can choose the geometric scaling center along the chord in the symmetry plane, and allow automatic re-positioning of the tailplanes along the aircraft's longitudinal axis in order to compensate for longer or shorter fuselages.

**Propulsion integration:** The capability to re-size predefined nacelle geometries has been added to the propulsion system integration module. If activated, the given geometry is linearly scaled with a factor determined by the ratio of scaled sea level static thrust (SLST) and original SLST of the unscaled engine.

Another new design option within the module is the over-the-wing placement of nacelles. Due to shielding effects of the wing, this option has been of particular interest with regard to aircraft designs optimized for low noise emissions.

**Landing gear design:** The landing gear geometry has been included into the library for the aircraft geometry components. Until then, the landing gear was considered as a single component with a total mass, which made a dedicated class unnecessary. Now the landing gear is composed of its single components, each having its own mass. This allows for a more analytical approach to calculate the landing gear mass using the material and volume information of the component.

### 4.3. Detailed design

In addition to the changes of the component sizing tools mentioned above, MICADO has been expanded in individual disciplines, which enables the (optional) detailed design of selected aircraft components and systems.

**Structural wing design:** For detailed studies of the effects of different wing geometries and their respective materials, a sophisticated method has been derived and can optionally be used in the design process. This method allows investigation of geometric and aeroelastic effects on both the wing's mass and structural performance; the latter is measured in terms of stability, strain, and buckling failure margins. For the latest assessments of the impact of composite materials and high aspect ratio wings on the aircraft performance using this method, the reader is referred to Elqatary [12].

**Systems design:** With regard to aircraft systems, the ability to design advanced systems for current technology levels has been enabled from internal projects [13]. Consequently, more-electric architectures using for instance electrical air conditioning, anti-icing, and electro-mechanical/electro-hydrostatic actuators can be designed in addition to conventional hydraulic or bleed air supply systems.

A more detailed description of the circuits of energy sources has also been implemented to consolidate individual consumers. This allows the individual consumers to be assigned to the respective circuits and the sizing of necessary pumps and ducts (hydraulic circuit) and/or generators and cables (electric circuit) [14].

**HLFC system design:** A significant extension of MICADO includes the possibility of designing an aircraft with integrated hybrid laminar flow control (HLFC). To demonstrate the aerodynamic benefit of this technology, a new method to predict (laminar) drag polars has been introduced (see Sec. 4.4). For a consistent aircraft design, however, a well-balanced complexity and prediction accuracy among the individual modules is required. Consequently, an HLFC system design methodology has been implemented to evaluate the potential of HLFC. Thus, not only the positive effects of the improved aerodynamics but also negative aspects, such as an increased empty mass and a decreased engine performance (due to necessary power offtakes), are considered. The current design concept is generally based on the simplified suction approach derived from the European project ALTTA for the A320 fin [15]. This concept is characterized by a perforated surface, suction chambers, and a suction plenum in the leading edge of the wing. The HLFC system itself is composed of various components, such as compressors, ducts, and electric motors. With pre-defined suction and pressure distributions for the design point, the HLFC system is sized within the automated MICADO design process; this is done using supplementary methods and equations proposed by Pe and Thielecke [16].

The output parameters include the HLFC system mass and the electrical power required to maintain the required suction flow. Through variations of the architecture topology, i.e. number of compressors and ducting architecture, the HLFC system can be further optimized. Recent developments of this tool mainly include an improved HLFC system architecture definition to meet current standards of HLFC architecture concepts [17].

### 4.4. Design analysis

Besides the changes for general aircraft sizing, the tools for design analysis have also been further developed. A short but not exhaustive overview of the major changes is given below.

**Mass estimation:** For the landing gear mass, an analytical method has been implemented in MICADO. This method replaces the formerly used semi-empirical equation of Thorbeck [18], which predicts the mass of the landing gear solely based on the maximum landing mass of the aircraft. The new approach uses the component-wise mass prediction from Schulz [19] and modifies it slightly, e.g., by considering further components like connection and support struts.

In addition to the improved mass prediction of conventional aircraft components, new methods have been derived to predict masses of electric powertrain components, such as batteries, superconductors, electric motors, and cryogenic cooling systems. This allows for detailed studies, e.g., on the influences of propulsion architectures with different degrees of hybridization. For a comprehensive overview of the various methods, the reader is referred to Aigner et al. [20].

**Aerodynamics:** To evaluate the application of Hybrid Laminar Flow Control (HLFC), a so-called quasi-three-dimensional (2.5D) approach has been developed. This approach is based on an iterative process interconnecting the 2D flow solver MSES with the 3D transition prediction module STABTOOL; this is realized by transformation rules for flow conditions and known wing geometries. Since the overall design of an aircraft does not only require individual points but rather the calculation of entire drag polars, a database approach has been implemented in MICADO. In this database, both turbulent and laminar drag polars previously calculated with the 2.5D approach are stored and can be accessed during the aircraft design process. This approach allows both laminar retrofits of turbulent aircraft and the optimization of pre-defined wing geometries. For more information on the basic approach, the reader is referred to Risse [1] and Risse et al. [21, 22]; in addition, a summarizing overview of the recent developments of the integrated HLFC assessment in MICADO can be found in Schültke et al. [17].

Another promising technique for improving aerodynamics during flight is the application of morphing devices such as variable cambers (VC) for the wing geometry. To investigate the potential of VC, a tool chain was set up using the conventional MICADO loop as well as more sophisticated tools to modify the airfoil geometries and subsequently analyze their aerodynamic behaviour; the latter uses the same database method as for the integration of the HLFC technology described before. For VC, however, drag polars of different airfoil permutations are merged into one VC polar. This follows the idea that the airfoil geometry can be changed during flight to adjust its optimum lift-to-drag ratio to the current lift coefficient. [23, 24]

For investigation of the effect of propellers installed upstream to the wing, the aerodynamics module has been extended to enable modeling of the corresponding aero-propulsive interaction. First, the propeller downwash for a range of flight and propeller operating conditions is calculated by using a combined blade element and momentum theory. Second, the impact of the propellers on wing aerodynamics is reduced to a dynamic pressure multiplier for each flight and propeller operating condition and stored in so-called prop-wing-interaction maps. Finally, the dynamic pressure multipliers are used to adjust lift and drag coefficients of the freestream polars to the flight and propeller operating conditions during mission simulation. [25]

To investigate the flight behavior and the drag reduction potential of two aircraft flying in formation, a tool chain was set up combining MICADO with external tools. These tools are the Athena Vortex Lattice Method (AVL) for determination of induced drag at the trailing aircraft, DATCOM+ Pro for the respective dynamic derivatives, and finally, the flight simulation program JSBSIM for the consideration of various atmospheric disturbances [26, 27, 28, 29, 30]. In this particular case, MICADO is used exclusively to create the aircraft data sets, which are then automatically processed in the tools mentioned above.

**Mission analysis:** As already described above, new propulsion concepts with different degrees of hybridization can now be considered within MICADO. In order to further enable the evaluation of any aircraft on the overall design level, the mission analysis was adapted to allow different thrust generators to be considered. Thus, it is possible to investigate the influence on both the fuel and energy required for a particular mission as well as their respective distribution.

Another feature that has been implemented in the mission analysis module is the opportunity to preset the ratio of time- and fuel-related costs. This so-called cost index can be used to calculate the most economical Mach number considering the current aircraft mass, altitude, and flight speed [31].

### 4.5. Evaluation

MICADO provides a number of post-processing analysis capabilities enabling investigation of aircraft designs based on a variety of metrics with respect to economic, ecological, and sociological aspects.

**Economic analysis:** A life cycle cost assessment methodology was introduced, including development, production, operation, and end-of-life of an aircraft [32, 33, 34, 35, 36]. Kreimeier extended the analyses for electric propulsion system components [37].

**Ecological analysis:** A full ecological assessment of the aircraft life cycle was also developed in the context of the PhD thesis by Schäfer [32]. The analyses do not primarily focus on the amount of exhaust emissions during the different phases of the life cycle, but rather on the environmental effect associated with these emissions. It is possible to evaluate the ecological impact of exhaust emissions with respect to different established climate metrics, such as Average Temperature Response and Absolute Global Warming Potential. [38]

**Sociological analysis:** Another subject raising ecological as well as sociological issues is aircraft noise. Noise assessment was first implemented in MICADO within a PhD thesis in 2016 [39]. The noise assessment module includes modeling of noise sources, noise propagation, and the psycho-acoustic effect on humans [40, 41, 42, 43, 44]. Since then, the noise module has been extended, for instance, to account for noise shielding due to specific placement of the engines [45, 46, 47, 48].

### 4.6. Framework

Besides the previously presented changes in the individual program modules, the overall design framework has been further developed as well. The convergence loop tool managing the design iteration process has been extended to perform an automatic pitch trimming of the aircraft in cruise conditions, thereby taking into account trim drag.

A second new feature of the overall convergence loop enables calibration towards already existing aircraft. In the so-called reference design mode, MICADO starts its calculations based on a given aircraft geometry (e.g. from drawings of the manufacturer) and fills in the gaps of the design by using the available program modules. Additionally, MICADO can calibrate the design to match the reference aircraft's operating mass empty and/or maximum take-off mass. The calibration can be performed by either a variation of the aerodynamic efficiency (aerodynamic calibration), or the specific fuel flow of the engines (engine calibration).

### 4.7. Changes for small conventional and electric aircraft

With the advent of electric flying, the conceptual design and evaluation of small aircraft certifiable according to EASA certification specification CS-23 has become increasingly relevant for the research community. Due to the generic design and modularity of the MICADO environment, on the one hand, and the similar characteristics and requirements of small and large aircraft, on the other hand, most of the existing tools could be adapted by just adding methods for conventional small aircraft. For example, the tool for aircraft component mass estimation was supplemented with semi-empirical relationships for component masses of small aircraft summarized by Gudmundsson [49]. Another example is the extension of the fuselage design tool to design typical fuselage shapes of small aircraft.

In contrast to the approach for small conventional aircraft, the design and evaluation of small electric aircraft required the development of new methods. The constraint analysis regarding wing-loading and power requirements for different phases of flight within the module for initial aircraft sizing was first adapted to small conventional aircraft and later extended for hybrid- and all-electric aircraft in order to initially estimate fuel and battery masses according to the degree of hybridization.

On conceptual aircraft design level, conventional turbofan engines can be modeled as a single component with negligible efficiency losses in the fuel system, such as power for fuel pumps. This is no longer true for all- and hybrid-electric propulsion architectures, which are characterized by a strong interdependence of the individual components. Therefore, the entire MICADO engine sizing module was re-designed to allow for detailed sizing of the individual powertrain components from propeller to battery; before, a linear scaling of predefined engine performance maps was adequate. A method to calculate the overall combined performance of propeller and piston engine or electric motor was implemented. Additionally, methods to design and integrate battery packs, either on pack- or cell-level, wiring, and power electronics were developed. For serial-hybrid architectures, a logic to size the range-extending internal combustion engine was added.

The integration of high-lift propellers is seen as one possibility to increase aerodynamic cruise efficiency due to higher wing-loading while, at the same time, maintaining low stall speeds as required for aircraft in the General Aviation class. To investigate this potential, models to design high-lift propellers and to simulate the aero-propulsive coupling of the high-lift propellers and the wing were developed and implemented.

A comprehensive summary of the changes to MICADO to model and simulate small conventional and electric aircraft can be found in the PhD thesis of Kreimeier [37]. In addition, Aigner et al. [50] give an overview of the design and the evaluation of hybrid-electric propulsion concepts for both small and large aircraft. Studies with MICADO on electric propulsion can be found in [20, 51, 52, 53].

### 4.8. Summary

The previously described advancements of the methods within the MICADO modules open up various new opportunities. Besides designing an aircraft from scratch with a given set of requirements, it is now additionally possible to re-design already existing aircraft. Moreover, the design space has been extended to smaller aircraft from the CS-23 sector. Newly introduced analysis programs taking into account economic, ecological, and sociological aspects allow for a holistic evaluation of aircraft designs. Furthermore, innovative technologies, such as novel propulsion concepts, more electric aircraft systems, and hybrid laminar flow control, can now be evaluated with MICADO. The following chapter describes the wide range of applications enabled by these advancements.

## 5. APPLICATION AREAS

Since the early beginnings, the MICADO environment has been used in various projects and for the academic teaching of students at RWTH Aachen University. Thereby, it has been possible to continuously improve and extend the capabilities of the software environment as well as to adapt it to technical innovations in the aviation sector on the one hand and software user requirements on the other hand. This chapter gives an overview of the application of MICADO in several projects, as well as the Central Reference Aircraft Data System (CeRAS), and the way MICADO is integrated into teaching.

### 5.1. Projects

The Institute of Aerospace Systems has strong ties with partners from industry and academia all over the world. Due to available funding schemes for academic research, however, most projects are carried out on national and European level. Projects that involve the MICADO environment usually feature a part that is intended to improve and extend both modeling and simulation capabilities, as well as a technology evaluation part, in which the added capabilities are used. The projects carried out in recent years can be divided into three groups according to the topics covered. The first group of projects gathers around the optimization of aircraft aerodynamics. Electrification of the aircraft powertrain is the common ground for the second group of projects. The third group of projects deals with topics other than the two mentioned before.

**Projects related to aerodynamics:** Aerodynamic optimization of aircraft has always been a key research interest in aerospace engineering. Against this background, it is self-evident that technology evaluation with regard to aircraft aerodynamics is one of the significant research pillars with MICADO.

Within the European project *FLEXOP*, the objective was to design a derivative aircraft with a high aspect-ratio wing to reduce fuel consumption by 7 % or increase payload by 20 % with minimizing certification costs. A methodology for static aero-elasticity with the ability to use the mass and stiffness matrices of carbon fiber reinforced polymer-wings was implemented to generate the flight shape from a wing's jig shape. [12]

Another European project called *SARISTU* addressed the integration of an adapting droop nose into the leading edge for a morphing wing. To accomplish the task, design rules were composed, a reference aircraft designed using conventional slats for comparison, and the system architecture was proposed. Finally, the morphing wing technology was evaluated on overall aircraft level. [54]

In the German LuFo-project *AVACON*, overall design studies are conducted that take into account the interdisciplinary coupling of the HLFC technology and over-the-wing ultra high-bypass engines. A medium-range reference aircraft with conventional under-the-wing engines is designed and analyzed first to be able to investigate the fuel savings potential later [55]. Further aircraft with different engine positions are then derived from this reference and subsequently evaluated.

In the project *BIMOD*, the technology of oscillating flaps is implemented in MICADO to investigate the potential for increasing the maximum high-lift coefficient. The evaluation is carried out for aircraft in the Airbus A350 class. [56]

**Projects related to electric propulsion:** Due to technological progress in the development of essential components for electric powertrains, electric aircraft propulsion has become more relevant. In contrast to conventional combustion engines, electric motors can be scaled more easily without having to accept significant efficiency losses. This opens the aircraft design space considerably, which consequently results in the need to investigate and understand this new design space. To date, projects for electric aircraft from the General Aviation class up to the class of short-range transport aircraft have been carried out.

In the context, the project *HyFly* focused on the conceptual development of a hybrid-electric powertrain for a Dornier Do-228. One of the key tasks was the design of an internal propulsion unit consisting of a high reluctance generator and a gas turbine. MICADO was used to evaluate the new design and to compare it against the reference configuration. [57]

Within the project *MVDC-OnBoard*, the focus is set on systems-integration, addressing the question on how to transfer energy efficiently and safely through the aircraft for large commercial airliners. Multiple energy networks with different voltage levels are considered as well as utilizing high-temperature superconductors (HTS) for power transmission. [20, 50]

The project *SAT*[7] puts the focus on aircraft from the General Aviation class. The goal is to identify suitable configurations for highly automated and sustainable transport of four passengers on regional connections of up to 500 km distance.

The latest project within the context of electric flying is *GNOSIS*[8] - a holistic analysis of electric flying that includes an evaluation of global emissions, local air

---

[7] https://ilr.rwth-aachen.de/go/id/ntyn

[8] https://ilr.rwth-aachen.de/go/id/jffpy

quality, costs, noise, and safety aspects within the context of the entire aircraft life cycle. The assessment is carried out for aircraft with 9 to 50 seats.

**Projects related to other topics:** The third group of MICADO-related projects deals with various topics and underlines the wide range of applications of this aircraft design and evaluation environment.

The European project *TeDiMo*[9] focused on technology diffusion modeling. MICADO was used to carry out parameter studies on the correlation between different component masses and fuel consumption or emissions.

*AGILE* is representative of projects in which only parts of the MICADO suite are used [58, 59, 60]. The tools for estimation of operating costs and exhaust gas emissions were integrated into DLR's Remote Component Environment [11]. Within the context of the follow-on project *AGILE 4.0*, a new maintenance module is developed and integrated into the MICADO environment to design aircraft following a maintenance-driven approach.

In 2017, the ILR prepared a report on the climate change mitigation potential through climate-optimized aircraft design for the German Environment Agency. After the integration of a model for the climate impact of aircraft emissions into the MICADO environment, parameter studies revealed that reduced flight speeds and altitudes are one option to reduce the climate impact of aviation. [61]

The *FORMIC* project investigated the potential of formation flight. For the evaluation, models were developed to calculate the direct operating costs, the passenger comfort, and to consider inefficiency when the aileron is used for trimming. Additionally, emergency procedures for flying in a formation have been developed. They include unplanned termination of the formation flight, subsequent return to the formation, or independent completion of the mission. [30]

In the *PAKO* project on psycho-acoustic aircraft optimization, models for the generation and propagation of noise at the aircraft structure and engine components were implemented in MICADO. This allows the evaluation of different engine positions and approach procedures concerning psycho-acoustic perception of aircraft noise. [62]

In addition to publicly funded research projects with and without partners from industry, MICADO is also used in projects in which partners from industry act as customers. These partners include both OEMs and suppliers of the aviation industry. In contrast to publicly funded research projects, which most often include major extensions to the modeling and simulation capabilities, the focus in projects with industry partners is rather on technology evaluation studies.

### 5.2. Central Reference Aircraft Data System CeRAS

CeRAS provides reference aircraft data and methods that can be used by the research community [63]. It is the outcome of a joint project of ILR and the Future Project Office (FPO) of Airbus. The aircraft database, as well as the design and evaluation methods, are open to the public and accessible online through a web page [64]. This web page is intended to serve as a living open-source platform, where the research community can communicate and contribute to. In a workshop with the German aircraft design community from industry and academia, the CeRAS approach was confirmed by all participants as a suitable common platform for application in future research projects.

The first reference aircraft in the database is a short-range aircraft called CSR-01. Regarding its top-level aircraft requirements, it is similar to an Airbus A320. The CSR-01 was designed and evaluated using the MICADO software environment. The publicly available data set comprises detailed technical aircraft design and economic as well as ecological assessment data. Plots and report files enrich the data. More reference aircraft, such as mid- and long-range aircraft, are to be added in the future.

### 5.3. Integration into teaching

The Institute of Aerospace Systems provides engineering students at RWTH Aachen University with basic and advanced knowledge on technology, systems, and design of aircraft. Students get in touch with MICADO within the context of courses and often when working on their final theses. While thesis projects include both MICADO code development and preparation of design studies, within the context of teaching, MICADO is only used to carry out aircraft design studies. As of late, students have the opportunity to investigate the impact of particular design decisions on aircraft performance with MICADO and present the results within the course *Aircraft Design II*. The outcome is graded and considered in the final course grade. The course *Computer-aided Aircraft Design* introduces the iterative character of the aircraft conceptual and preliminary design process in more detail. In addition to theoretical basics, the students learn about the MICADO-specific program structure for the design and evaluation of aircraft. Afterward, the course's focus is on the methodology for conducting parameter studies to find optimized aircraft designs.

## 6. OUTLOOK

Within this publication, the status of MICADO in August 2020 was presented. Starting from this date, for the next few years the following improvements and extensions are planned:

**Software architecture:** The process for code standardization will be improved and extended in a way that code is further extracted to libraries as far as possible down to function level. In addition, the developer documentation will be consistently continued and interconnections between functions and tools will be visualized. All this will be done in preparation

---

[9] https://ilr.rwth-aachen.de/go/id/izpio

for a guided aircraft design process, using a kind of artificial intelligence. For code testing, first, the MICADOcheck module will be extended enabling to compare an aircraft design created with a modified MICADO version to an already existing aircraft design on the repository. Second, processes and tests will be implemented at the module and function level (unit tests). As a second instance for code verification and validation, a review process will be defined and applied.

For MICADO users, a user documentation will be written to further facilitate the entry into the application of MICADO. Moreover, the GUI will be further developed considering user feedback.

**MICADO methodology:** In terms of MICADO methodology, the level of detail of modules and calculation methods will be further enhanced and new technologies will be integrated, e.g. models for engines using biofuels.

**Application:** In projects, MICADO will be mainly used as a technology evaluator. Besides, MICADO will be used to perform lots of aircraft variations and studies to collect data. This data will be collected and stored to be reusable in preparation for a self-learning aircraft design environment using artificial intelligence.

Finally, the reference aircraft data system CeRAS will be extended starting with a version of a mid-range aircraft (CMR-01). This reference aircraft will be based on the *AVACON* reference aircraft which has already been validated by Airbus.

## Acknowledgements

The authors would like to thank all former and current members of the MICADO project group of the Institute of Aerospace Systems (ILR) at RWTH Aachen University for their excellent collaboration for over more than a decade. Without their contributions, the software could not have been developed in the same way that it was. We would like to thank especially the former colleagues, who have dedicated their time at ILR to the development of MICADO: Eckhard Anton, Michael Kreimeier, Tim Lammering, Nils Lange, Yaolong Liu, Miguel Pereda Albarrán, Fabian Peter, Kristof Risse, Abhishek Kumar Sahai, and Katharina Schäfer. Furthermore, we thank all students involved for their outstanding work as student assistants and/or their valuable project, bachelor, and master theses.

## Contact address

florian.schueltke@ilr.rwth-achen.de

## References

[1] K. Risse. “Preliminary Overall Aircraft Design with Hybrid Laminar Flow Control”. PhD thesis. RWTH Aachen University, 2016. DOI: `10.18154/RWTH-2017-00974`.

[2] K. Risse et al. “An Integrated Environment for Preliminary Aircraft Design and Optimization”. In: *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference: SciTech 2012*. American Institute of Aeronautics and Astronautics, 2012. DOI: `10.2514/6.2012-1675`.

[3] K. H. Horstmann. *Ein Mehrfach-Traglinienverfahren und seine Verwendung für Entwurf und Nachrechnung nichtplanarer Flügelanordnungen*. DFVLR-FB 87-51. Institut für Entwurfsaerodynamik der DFVLR, Braunschweig, 1987.

[4] S. Le Digabel. “Algorithm 909: NOMAD: Nonlinear Optimization with the MADS Algorithm”. In: *ACM Transactions on Mathematical Software* 37.4 (2011), pp. 44:1–44:15. DOI: `10.1145/1916461.1916468`.

[5] R. C. Martin et al. *Clean code. A handbook of agile software craftsmanship*. Robert C. Martin series. Upper Saddle River, NJ et al.: Prentice Hall, 2009. ISBN: 978-0132350884.

[6] F. Schueltke and E. Stumpf. “UNICADO: Aufbau und Etablierung einer universitären Flugzeugvorentwurfsumgebung”. In: *69. Deutscher Luft- und Raumfahrtkongress*. Virtual event: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2020.

[7] B. Nagel et al. “Communication in aircraft design: Can we establish a common language?” In: *28th International Congress of the Council of the Aeronautical Sciences*. Edinburgh: Optimage Ltd. on behalf of the International Council of the Aeronautical Sciences, 2012. ISBN: 978-0-9565333-1-9.

[8] Deutsches Zentrum für Luft- und Raumfahrt e.V. *TiGL: Open Source Computational Geometry Library*. Simulation and Software Technology, Intelligent and Distributed Systems. URL: `https://dlr-sc.github.io/tigl/` (visited on 08/26/2020).

[9] M. Siggel et al. “TiGL: An Open Source Computational Geometry Library for Parametric Aircraft Design”. In: *Mathematics in Computer Science* 13.3 (2019), pp. 367–389. ISSN: 1661-8270. DOI: `10.1007/s11786-019-00401-y`.

[10] Deutsches Zentrum für Luft- und Raumfahrt e.V. *TiXI: Open Source XML Interface Library*. Simulation and Software Technology, Intelligent and Distributed Systems. URL: `https://github.com/DLR-SC/tixi` (visited on 08/26/2020).

[11] B. Boden et al. *RCE: An Integration Environment for Engineering and Science*. 2019. URL: `https://arxiv.org/pdf/1908.03461` (visited on 08/26/2020).

[12] I. Elqatary, F. Schueltke, and E. Stumpf. “Geometrical and Weight Wing Optimization Impact on the Performance of Civil Transport Aircraft”. In: *69. Deutscher Luft- und Raumfahrtkongress*. Virtual event: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2020.

[13] T. Lammering. “Integration of Aircraft Systems into Conceptual Design Synthesis”. PhD thesis. RWTH Aachen University, 2014. URL: `https://publications.rwth-aachen.de/record/229107/files/5048.pdf` (visited on 08/26/2020).

[14] B. Aigner et al. “Assessment of Electromechanical Flight Control Actuators with Regard to Direct Operating Costs”. In: *65. Deutscher Luft- und Raumfahrtkongress*. Braunschweig: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2016.

[15] K. H. Horstmann et al. “A Simplified Suction System for an HLFC Leading Edge Box of an A320 Fin”. In: *CEAS Aerospace Aerodynamics Research Conference*. Cambridge: Council of European Aerospace Societies, 2002.

[16] T. Pe and F. Thielecke. “Synthesis and Topology Study of HLFC System Architectures in Preliminary Aircraft Design”. In: *3rd CEAS Air & Space Conference*. Venice: Council of European Aerospace Societies, 2011.

[17] F. Schueltke, T. Effing, and E. Stumpf. “MICADO: Recent Developments within the Preliminary Overall Aircraft Design with Integrated Hybrid Laminar Flow Control: Präsentation”. In: *69. Deutscher Luft- und Raumfahrtkongress*. Virtual event: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2020.

[18] J. Thorbeck. *Manuskript zur integrierten Lehrveranstaltung Flugzeugentwurf I und II*. TU Berlin: Institut für Luft- und Raumfahrt, Fachgebiet Luftfahrzeugbau und Leichtbau, 2003.

[19] A. J. Schulz, U. B. Carl, and C. Koeppen. “Entwicklung eines Prognosemodells für Masse und Leistungsbedarf von Fahrwerken im Flugzeugvorentwurf”. In: *53. Deutscher Luft- und Raumfahrtkongress*. Dresden: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2004. DOI: `10.15480/882.211`.

[20] B. Aigner, M. Nollmann, and E. Stumpf. “Design of a Hybrid Electric Propulsion System within a Preliminary Aircraft Design Environment”. In: *67. Deutscher Luft- und Raumfahrtkongress*. Friedrichshafen: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2018.

[21] K. Risse et al. “Conceptual Wing Design Methodology for Aircraft with Hybrid Laminar Flow Control”. In: *52nd Aerospace Sciences Meeting*. National Harbor, Maryland: American Institute of Aeronautics and Astronautics, 2014. DOI: `10.2514/6.2014-0023`.

[22] K. Risse and E. Stumpf. “Conceptual aircraft design with hybrid laminar flow control”. In: *CEAS Aeronautical Journal* 5.3 (2014), pp. 333–343. DOI: `10.1007/s13272-014-0111-6`.

[23] F. Peter et al. “Economic assessment of morphing leading edge systems in conceptual aircraft design”. In: *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. Grapevine, Texas: American Institute of Aeronautics and Astronautics, 2013. DOI: `10.2514/6.2013-145`.

[24] F. N. Peter et al. “Variable Camber Impact on Aircraft Mission Planning”. In: *53rd AIAA Aerospace Sciences Meeting 2015*. Vol. AIAA 2015-1902. Kissimmee, Florida: American Institute of Aeronautics and Astronautics, 2015. DOI: `10.2514/6.2015-1902`.

[25] M. Kreimeier et al. “Economic Assessment of Air Mobility On-Demand Concepts”. In: *Journal of Air Transportation* 26.1 (2018), pp. 23–36. DOI: `10.2514/1.D0058`.

[26] Y. Liu et al. “Assessment of Potential Benefit of Formation Flight at Preliminary Aircraft Design Level”. In: *53rd AIAA Aerospace Sciences Meeting 2015*. Kissimmee, Florida: American Institute of Aeronautics and Astronautics, 2015. DOI: `10.2514/6.2015-1907`.

[27] Y. Liu and E. Stumpf. “Investigation on Passenger Comfort of Formation Trailing Aircraft at Different Atmospheric Turbulence Levels”. In: *64. Deutscher Luft- und Raumfahrtkongress*. Rostock: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2015.

[28] Y. Liu. “Investigation on the Benefit and Feasibility of Applying Formation Flight to Civil Transport Aircraft”. PhD thesis. RWTH Aachen University, 2016.

[29] Y. Liu and E. Stumpf. “Variable Camber Application to Aircraft in Formation Flight”. In: *54rd AIAA Aerospace Sciences Meeting 2016*. San Diego, California: American Institute of Aeronautics and Astronautics, 2016. DOI: `10.2514/6.2016-0773`.

[30] C. Zumegen and E. Stumpf. “Flight behavior of long-haul commercial aircraft in formation flight”. In: *23rd ATRS World Conference*. Amsterdam: Air Transport Research Society, 2019.

[31] F. Schueltke et al. “Auswirkungen der Vorgabe eines Kostenindex auf den Flugzeugentwurf”. In: *66. Deutscher Luft- und Raumfahrtkongress 2017*. München: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V. URL: `https://www.dglr.de/publikationen/2018/450261.pdf` (visited on 08/26/2020).

[32] K. Schäfer. “Conceptual Aircraft Design for Sustainability”. PhD thesis. RWTH Aachen University, 2018. ISBN: 978-3-8440-5955-7.

[33] K. Franz et al. “Life Cycle Engineering in Preliminary Aircraft Design”. In: *Leveraging Technology for a Sustainable World*. Ed. by D. A. Dornfeld and B. S. Linke. Berlin: Springer, 2012, pp. 473–478. DOI: `10.1007/978-3-642-29069-5`.

[34] K. Franz et al. “Economics of Laminar Aircraft Considering Off-Design Performance”. In: *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*. Honolulu, Hawaii: American Institute of Aeronautics and Astronautics, 2012. DOI: `10.2514/6.2012-1760`.

[35] K. Franz et al. “Interdisziplinäre Bewertungsplattform zur Lebenszyklusanalyse im Flugzeugvorentwurf”. In: *62. Deutscher Luft- und Raumfahrtkongress*. Stuttgart: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2013.

[36] K. Franz, K. Risse, and E. Stumpf. “Framework for Sustainability-Driven Aircraft Design”. In: *2013 Aviation Technology, Integration, and Operations Conference*. Los Angeles, California: American Institute of Aeronautics and Astronautics, 2013. DOI: `10.2514/6.2013-4393`.

[37] M. Kreimeier. “Evaluation of On-Demand Air Mobility with Utilization of Electric Powered Small Aircraft”. PhD thesis. RWTH Aachen University, 2018. URL: `https://publications.rwth-aachen.de/record/722626/files/722626.pdf` (visited on 08/26/2020).

[38] M. Kreimeier and K. Franz. “Nachhaltigkeitsbewertung des Flugbetriebs für den Flugzeugvorentwurf”. In: *62. Deutscher Luft- und Raumfahrtkongress*. Stuttgart: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2013. URL: `https://www.dglr.de/publikationen/2014/301483.pdf` (visited on 08/26/2020).

[39] A. K. Sahai. “Consideration of Aircraft Noise Annoyance during Conceptual Aircraft Design”. PhD thesis. RWTH Aachen University, 2016. URL: `https://publications.rwth-aachen.de/record/668901/files/668901.pdf` (visited on 08/26/2020).

[40] A. Sahai et al. “Interdisciplinary Auralization of Take-off and Landing Procedures for Subjective Assessment in Virtual Reality Environments”. In: *18th AIAA/CEAS Aeroacoustics Conference (33rd AIAA Aeroacoustics Conference)*. Colorado Springs, Colorado: American Institute of Aeronautics and Astronautics, 2012. DOI: `10.2514/6.2012-2077`.

[41] A. Sahai and E. Stumpf. “A Comparative Analysis of the Subjective Assessment of Standard and Noise Abatement Aircraft Procedures”. In: *National Conference on Noise Control Engineering*. Denver, Colorado: Institute of Noise Control Engineering, 2013.

[42] A. Sahai and E. Stumpf. “Incorporating and Minimizing Aircraft Noise Annoyance During Conceptual Aircraft Design”. In: *20th AIAA/CEAS Aeronautics Conference*. AIAA Aviation Forum. Atlanta, Georgia: American Institute of Aeronautics and Astronautics, 2014. DOI: `10.2514/6.2014-2078`.

[43] A. Sahai et al. “Interactive simulation of aircraft noise in aural and visual virtual environments”. In: *Applied Acoustics* 101 (2016), pp. 24–38. DOI: `10.1016/j.apacoust.2015.08.002`.

[44] A. K. Sahai, M. Snellen, and D. G. Simons. “Aircraft Design Optimization for Lowering Community Noise Exposure Based on Annoyance Metrics”. In: *Journal of Aircraft* 54.6 (2017), pp. 2257–2269. DOI: `10.2514/1.C034009`.

[45] M. Pereda Albarrán, A. K. Sahai, and E. Stumpf. “Aircraft noise sound quality evaluation of continuous descent approaches”. In: *46th International Congress and Exposition on Noise Control Engineering*. Hong Kong: Institute of Noise Control Engineering, 2017.

[46] M. Y. Pereda Albarrán, F. Schueltke, and E. Stumpf. “Sound Quality Assessments of Over-the-Wing Engine Configurations Applied to Continuous Descent Approaches”. In: *AIAA Aviation Forum*. Atlanta, Georgia: American Institute of Aeronautics and Astronautics, 2018. DOI: `10.2514/6.2018-4083`.

[47] M. Y. Pereda Albarrán and E. Stumpf. “Parametric Aircraft Configuration Optimization according to ICAO Annex 16 Certification Standards and Sound Quality Evaluation within Conceptual Aircraft Design”. In: *The 47th International Congress and Exposition on Noise Control Engineering*. Chicago, Illinois: Institute of Noise Control Engineering, 2018.

[48] M. Nuño et al. “A Calculation Method for Over-the-Wing Engines within Conceptual Aircraft Design”. In: *67. Deutscher Luft- und Raumfahrtkongress*. Friedrichshafen: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2018. DOI: `10.25967/480177`.

[49] S. Gudmundsson. *General aviation aircraft design: Applied methods and procedures*. 1. edition. Oxford, UK: Butterworth-Heinemann, 2014. ISBN: 978-0-12-397308-5.

[50] B. Aigner et al. "An Integrated Design Framework for Aircraft with Hybrid Electric Propulsion". In: *AIAA Scitech 2020 Forum*. Orlando, Florida: American Institute of Aeronautics and Astronautics, 2020. DOI: `10.2514/6.2020-1501`.

[51] P. Strathoff and E. Stumpf. "Design Space Exploration for Small Aircraft with Hybrid-Electric Powertrains and Distributed Electric Propulsion". In: *67. Deutscher Luft- und Raumfahrtkongress*. Friedrichshafen: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2018. URL: `https://www.dglr.de/publikationen/2019/480183.pdf` (visited on 08/26/2020).

[52] P. Strathoff et al. "A Study on "Through-the-Road"-Parallel-Hybrid Powertrains for Small Aircraft with Distributed Electric Propulsion". In: *AIAA Aviation 2019 Forum*. Dallas, Texas: American Institute of Aeronautics and Astronautics, 2019. DOI: `10.2514/6.2019-3677`.

[53] P. Strathoff and E. Stumpf. "Performance Comparison of Conventional, Hybrid-Electric, and All-Electric Powertrains for Small Aircraft". In: *AIAA Aviation 2020 Forum*. Virtual event: American Institute of Aeronautics and Astronautics, 2020. DOI: `10.2514/6.2020-3252`.

[54] F. Peter et al. "Morphing Value Assessment on Overall Aircraft Level". In: *Smart Intelligent Aircraft Structures (SARISTU)*. Ed. by P. C. Wölcken and M. Papadopoulos. Cham: Springer International Publishing, 2016, pp. 859–871. DOI: `10.1007/978-3-319-22413-8`.

[55] T. Effing. "HLFC-optimierter Retrofit-Entwurf einer Mittelstrecken-Referenzkonfiguration im Rahmen des AVACON-Projekts: Präsentation". In: *68. Deutscher Luft- und Raumfahrtkongress*. Darmstadt: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2019.

[56] R. Stephan et al. "Influence of Dynamic Flap Movement on Maximum Lift and Wake Vortex Evolution". In: *68. Deutscher Luft- und Raumfahrtkongress*. Darmstadt: Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal - Oberth e.V., 2019. DOI: `10.25967/490118`.

[57] J.-P. Hofmann et al. "A Comprehensive Approach to the Assessment of a Hybrid Electric Powertrain for Commuter Aircraft". In: *AIAA Aviation 2019 Forum*. Dallas, Texas: American Institute of Aeronautics and Astronautics, 2019. DOI: `10.2514/6.2019-3678`.

[58] P. D. Ciampa and B. Nagel. "Streamlining Cross-Organizational Aircraft Development Processes: an Overview of the AGILE Project". In: *31st Congress of the International Council of the Aeronautical Sciences*. Belo Horizonte: International Council of the Aeronautical Sciences, 2018. ISBN: 978-3-932182-88-4.

[59] P. Della Vecchia et al. "Advanced Turboprop Multidisciplinary optimization within AGILE project". In: *AIAA Aviation 2018 Forum*. Atlanta, Georgia: American Institute of Aeronautics and Astronautics, 2018. DOI: `10.2514/6.2018-3205`.

[60] F. Torrigiani et al. "Design of the Strut Braced Wing aircraft in the AGILE Collaborative MDO Framework". In: *31st Congress of the International Council of the Aeronautical Sciences*. Belo Horizonte: International Council of the Aeronautical Sciences, 2018. ISBN: 978-3-932182-88-4.

[61] M. Husemann et al. *A Summary of the Climate Change Mitigation Potential through Climate-Optimized Aircraft Design*. Institute of Aerospace Systems (ILR), RWTH Aachen University, Aachen, 2017. DOI: `10.18154/RWTH-2019-02518`.

[62] M. Y. Pereda Albarrán and E. Stumpf. "Aircraft Engine Positioning Psychoacoustic Optimization within Conceptual Aircraft Design". In: *The 48th International Congress and Exhibition on Noise Control Engineering*. Madrid: International Institute of Noise Control Engineering, 2019.

[63] K. Risse et al. "Central Reference Aircraft data System (CeRAS) for research community". In: *CEAS Aeronautical Journal* 7.1 (2016), pp. 121–133. DOI: `10.1007/s13272-015-0177-9`.

[64] CeRAS. *Central Reference Aircraft data System*. Institute of Aerospace Systems (ILR), RWTH Aachen University, Aachen. URL: `https://ceras.ilr.rwth-aachen.de` (visited on 08/26/2020).