



MODELICA

Proceedings
of the 4th International Modelica Conference,
Hamburg, March 7-8, 2005,
Gerhard Schmitz (editor)

B. Oehler
Airbus Deutschland GmbH, Germany
**Modeling and Simulation of Global Thermal and Fluid Effects in an
Aircraft Fuselage**
pp. 497-506

Paper presented at the 4th International Modelica Conference, March 7-8, 2005,
Hamburg University of Technology, Hamburg-Harburg, Germany,
organized by The Modelica Association and the Department of Thermodynamics, Hamburg University
of Technology

All papers of this conference can be downloaded from
<http://www.Modelica.org/events/Conference2005/>

Program Committee

- Prof. Gerhard Schmitz, Hamburg University of Technology, Germany (Program chair).
- Prof. Bernhard Bachmann, University of Applied Sciences Bielefeld, Germany.
- Dr. Francesco Casella, Politecnico di Milano, Italy.
- Dr. Hilding Elmqvist, Dynasim AB, Sweden.
- Prof. Peter Fritzson, University of Linkping, Sweden
- Prof. Martin Otter, DLR, Germany
- Dr. Michael Tiller, Ford Motor Company, USA
- Dr. Hubertus Tummescheit, Scynamics HB, Sweden

Local Organization: Gerhard Schmitz, Katrin Pröhl, Wilson Casas, Henning Knigge, Jens Vassel,
Stefan Wischhusen, TuTech Innovation GmbH

Modeling and Simulation of Global Thermal and Fluid Effects in an Aircraft Fuselage

Dipl.-Ing. Bettina Oehler
Air Systems Technology and Processes
Airbus Deutschland GmbH

Abstract

This paper presents a model to determine pressure, temperature and humidity in designated compartments of an aircraft fuselage based on the evaluation of mass and heat flow balances under various environmental and operational boundary conditions. A library has been developed, containing all necessary modules and aircraft templates for modeling different aircraft types. Based on an aircraft template the end-user establishes a customer cabin layout, specifies simulation and operation conditions and evaluates the results. The program code, the library and the application of the template is easily understandable and structured to minimize errors.

This paper briefly describes how the Modelica modeling language was used for the calculation of the fluid properties and the solution of the mass and energy balances. Furthermore the implementation of an aircraft system is shown with a focus on data management. Improvement potential for the use of Dymola/Modelica for this type of application is identified.

Keywords: Thermal and Fluid Effects; Aircraft Fuselage Model; Thermodynamic Library.

1 Introduction

Consideration of the comfort felt by passengers and crew is of great importance in the design phase of air-conditioning system for an aircraft. Currently there are no agreed common standards for thermal comfort in the aircraft cabin, due to insufficient knowledge of the interaction of several parameters such as temperature, air speed and air humidity. These parameters all influence thermal comfort.

In addition, customers increasingly demand variety and flexibility of cabin layouts. Special compartments for example crew rest compartments or special

customized installations in the cabin, such as coffee bars, stair-houses and beauty salons, can be installed at various locations in the aircraft cabin.

In order to obtain a better understanding of the thermal and fluid effects in the whole fuselage a highly detailed simulation model has been developed by Airbus, named Overall Fuselage Flow Model (OFFM).

This model will support the work being carried out in the subproject Individual Seat Climatisation (ISC) of the research project Friendly Aircraft Cabin Environment (FACE). For the ISC subproject realistic boundary conditions are needed. The aim of ISC is to optimize the percentage of people that are satisfied with thermal comfort in aircraft cabins.

The simulation model is intended to provide boundary conditions for detailed CFD simulations.

The tool ultimately selected, Dymola/Modelica, best fulfills the tool requirements, such as connecting elements to networks, managing a high number of components without increasing complexity, available flow and heat libraries and accessibility to source code. For the simulation Modelica version 2.1 and Dymola version 5.3a are used.

2 Model Description

The modeled fuselage consists of all compartments inside the pressurized fuselage. The model boundaries related to the mass flows are the total air inlet into the mixer unit and the total air outlet flow of the fuselage. The temperature boundaries are the air temperature at the compartments' air outlets i.e. in the cabin, flight deck, forward and aft cargo compartment and temperatures for all adjacent, unpressurized compartments, for example main and nose landing gear, wing box and the skin temperature.

2.1 Description of Aircraft Air-Conditioning Systems

The environmental control system (ECS) supplies conditioned air to maintain the desired temperatures, ventilation rates and pressurization within the cabin zones and flight deck.

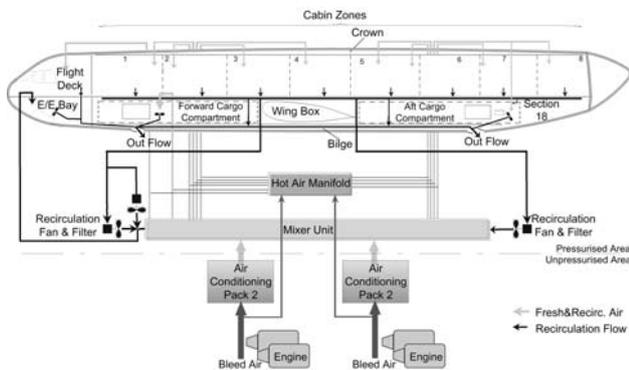


Figure 1: General Design of Environmental Control System

The ECS consists of the air-conditioning packs, the mixer unit, the air distribution/recirculation system and the pressurization system, including the associated fans. These different systems are shown in figure 1 for the A340-500/600.

The pneumatic system provides hot, pressure regulated bleed air from the engines (or Auxiliary Power Unit APU or external sources) to the air-conditioning packs.

Two air-conditioning packs, consisting of heat exchangers, compressor, water extraction and turbine, located in the unpressurized area below the center wing box provide conditioned air to the mixer unit.

In the mixer unit, installed under the cabin floor in front of the center wing box, the conditioned air is mixed with recirculated and filtered cabin air (extracted out of the cabin at floor level through dado panels), which is fed by recirculation fans to the mixer unit.

The air from the mixer unit is distributed to the cabin temperature control zones (for the A340-500/600 eight cabin areas are defined), the flight deck and cargo compartments. The independent temperatures for the different compartments can be achieved by directing hot trim air from the hot air manifold to the distribution ducts. In addition ventilation is provided for the electrical and electronic racks (E/E).

The cabin pressure is regulated automatically by the outflow valve position depending on the required cabin pressure change rate and differential pressure between cabin and ambient.

2.2 Conceptual Model Design

The overall fuselage flow model (OFFM) determines air mass flows, pressures, heat flows, temperatures and humidity of the entire aircraft fuselage for different ground and flight conditions.

For the one dimensional mass and heat flow calculation, the fuselage volume is divided into individual control volumes, such as flight deck, cabin, crown, E/E bay, cargo compartment, etc., see figure 2 and figure 3.

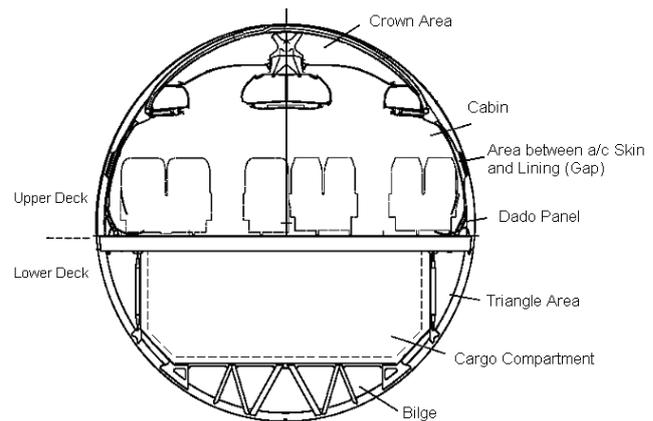


Figure 2: Aircraft Compartments in the Fuselage – Cross Section Overview

The modules include different thermo-fluid components, which cover the physical/mathematical description for the mass and heat flow within the fuselage.

The fuselage airflows are simulated by calculating the condition of the air within, and the pressure loss between these discrete volumes. For the heat flows all relevant heat sources and sinks, the heat exchange with adjacent control volumes and enthalpy flows are considered.

The release of humidity through metabolic processes is considered depending on the passenger load and in relation to the outside air flow per occupant provided to the cabin.

The simulated aircraft model consists of a federation of aircraft modules and overlaying system modules.

The aircraft modules are divided into fuselage standard modules and layout modules. Fuselage standard modules consider all aircraft modules, which are layout-independent. For example the following modules do not differ for one aircraft type with different system and cabin layouts: crown, triangle area, E/E bay, bilge, etc.

The cabin area is subject to various layouts such as different galley/lavatory positions and seat layouts (first, business and economy class). Nevertheless

several cabin area modules or parameters are layout-independent such as ceiling air outlets, cabin length, fuselage diameter, doors and the zone classification.

Consequently the specific layout modules are cabin installations such as galley, lavatories, seat rows, staircases and the utilization of the lower deck compartments. The whole or partial volume of the cargo compartments can be occupied e.g. with crew rest compartments.

The system modules contain the air supply and air extraction system, the cabin pressurization system and several individual fans. An example for the system implementation is given in detail for the air supply system in chapter 5.

The interaction between the system and the aircraft standard modules or specific layout modules is considered in two different ways; either by port connections or by a value assignment via variable names. No additional inputs in the system modules by the user are necessary.

With the aircraft standard modules, layout specific modules and the system modules a specific aircraft layout can be built, as shown in figure 3.

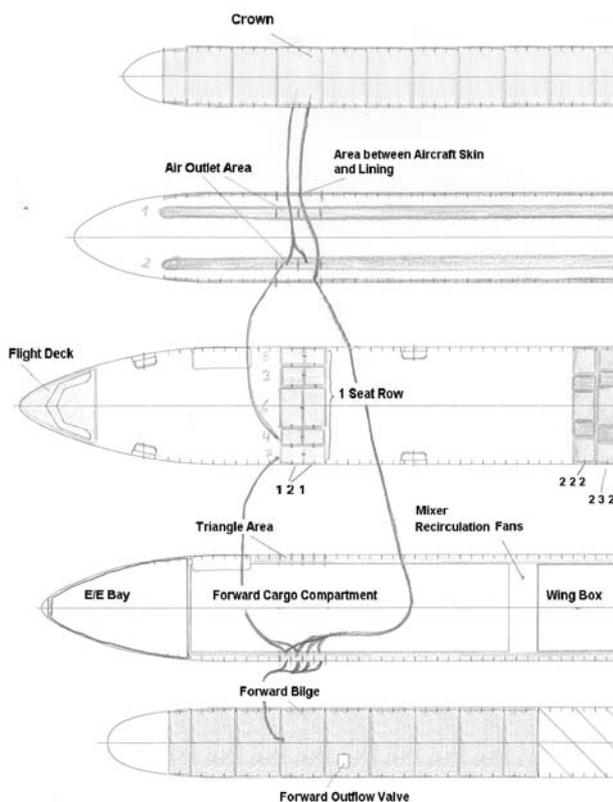


Figure 3: Principle Aircraft Template – Top View

Therefore a template is provided by the model developer containing all fuselage standard modules, system modules and an User Input Parameter Window (parameter dialog). This aircraft template is the basis for the user to create a new layout by implementing the layout specific modules and entering the

input values and boundary conditions for the simulation for different cabin, system and environmental conditions. The data management of the model in detail is described in chapter 4.

The simulation model offers the possibility of calculating

- isothermal or non-isothermal conditions,
- with dry or humid air and
- with different flow resistance laws.

For this application only the steady state results are of interest.

The implementation of the above described system and compartment modules results in a large and complex model, so that the model must be modular, adaptable and user-friendly. It is therefore important that the developer and the end-users can easily handle the model.

2.3 Library Structure

A special library is developed to establish a model of the fuselage. It consists of the control volume, the basic components for flow and heat such as flow and heat resistances, constant pressure sources/sinks and the property models for dry and humid air. Derived from these general basic elements a specific library section was created containing the aircraft modules.

One of the most important principles for the development of the library was simplicity, comprehensibility and reusability. In this respect the ThermoFluid [1] library was proofed to be too complicated. Moreover the Modelica_Fluid and Modelica_Media were not available when starting this project.

The OFFM library contains 6 main packages. Each package contains sub-packages, as described below.

1. General
 - Additional Units,
 - Functions,
 - Parameter,
 - Constant and
 - General Material Properties.
2. Icons - Icon Definitions
3. Interfaces - Flow and Heat Ports
4. Components
 - Control Volume,
 - Mass Flow Source,
 - Pressure Source,
 - Pressure Loss,
 - Heat Source,
 - Const. Temperature and
 - Heat Resistances.

5. Media – Dry Air and Humid Air
6. Aircraft_Models - contains packages for each aircraft type with the following sub-packages:
 - aircraft specific Data (Flow/Heat/Boundary Conditions),
 - layout-independent modules (aircraft standard modules and the corresponding assemblies),
 - aircraft complete (aircraft template) and
 - layout-dependent modules (Cabin, Lower Deck and System).

3 Implementation of Physical-Mathematical Description

3.1 Mass and Energy Balances

For the mass balance all internal mass flow sources and mass flows resulting from the control volumes differential pressures must be identified. Each control volume contains one definite flow port at which the sum-to-zero equations for all incoming and outgoing masses are automatically generated, due to the `flow` prefix. In the same flow port the enthalpy flows are determined correspondingly.

Additionally each control volume contains one definite heat port, where all heat flows resulting from internal heat sources and all heat flows transferred across the borders of the control volume are summed.

In the control volume the average value of pressure and temperature is calculated by solving the mass and energy balances, as depicted in figure 4 (the equations in the control volume are completed by humidity here not shown).

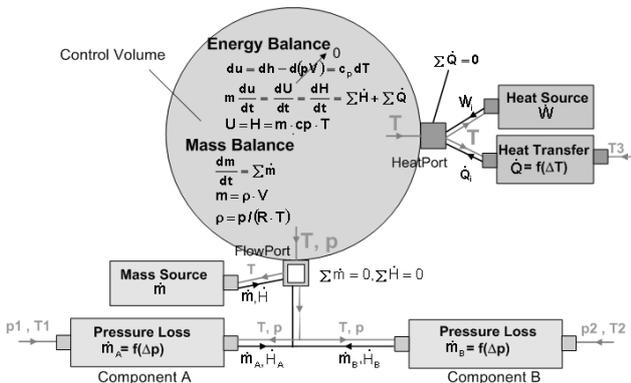


Figure 4: Mass and Energy Balances

The models are developed in such a way that it is possible to run a simulation under isothermal or non-isothermal conditions. The variable `includeEnergyBalance` is Boolean and can be set to true or

false, depending on whether the energy balance is used or not. The Modelica implementation is as follows:

```

...
if (includeEnergyBalance) then
  der (H) = FlowPort.H_dot + Heat
    Port.Q_dot;
  H = if (includeHumidAir) then
    humidAir.cp*m*T)
    else (dryAir.cp*m*T);
else
  der (H) = 0;
  T = T_0;
end if;
...
    
```

3.2 Pressure Loss

The pressure loss between the control volumes is described by the following equation:

$$\Delta p = k_1 \cdot \dot{m} + k_2 \cdot \dot{m} \cdot |\dot{m}|, \tag{3.1}$$

with k_1 = Linear Flow Coefficient,
 k_2 = Quadratic Flow Coefficient.

k_1 and k_2 are calculated values based on flight test evaluation.

For the calculation of the mass flow the following equation is implemented:

$$\dot{m} = fdir \cdot \left(\frac{1}{k_2} \right) \cdot \left[-\frac{k_1}{2} \cdot \text{ThermoRootFct} \sqrt{\left(\frac{k_1}{2} \right)^2 + k_2 \cdot |\Delta p|} \right] \tag{3.2}$$

with `fdir` as an indicator of the flow direction, it can reach the value 1 for design flow and -1 reversal flow,

the `thermo root` function from the free library `ThermoFluid` [1], with a linear interpolation near 0.

The scale of the parameters is given in the following: The flow resistance values varies between 10^2 - 10^5 Pa·s²/kg² depending on there location, whereas the pressure difference between the control volumes is in a range of 0.1 Pa up to 10 Pa which results to mass flows from 0.001 kg/s up to 0.01 kg/s.

The leakage flow from aircraft fuselage to ambient is calculated by the pressure vessel equation:

$$\dot{m}_{ij} = A_{eff} \cdot \sqrt{\frac{2p_i^2}{R \cdot T_i}} \cdot \sqrt{\frac{\kappa}{\kappa - 1} \left[\left(\frac{p_j}{p_i} \right)^{2/\kappa} - \left(\frac{p_j}{p_i} \right)^{(\kappa+1)/\kappa} \right]} \tag{3.3}$$

3.3 Fluid Properties

In the following section the calculation principle of the fluid properties for dry and humid air is described. The package `Media` contains the following classes separated for dry and humid air:

- Declaration and initialization of the fluid property variables such as specific enthalpy, thermal conductivity, dynamic viscosity, gas constant and specific heat capacity in a `record` and
- a `block` containing inputs and outputs and an equation part. The input of the block for dry air is the temperature and for humid air additionally the pressure and water content. The outputs are the property values. The equation part contains the empirical correlation based on [2] to calculate the fluid properties as a function of inputs.

The `block` class is used instead of a function in order to avoid differentiation of functions.

In the following paragraphs the interaction of the above referenced `block` and `record` class is described for dry air. Each model, where the fluid properties are needed, contains an instance of the `record` and `block` in the declaration part in the following way:

```
OFFM.Media.Dry_Air.DryAirProp
  DryAir(T=T);
OFFM.Media.Dry_Air.DryAirRec dryAir=
  DryAir.DryAirRec;
```

In the `block` class `DryAir` the values of the fluid property variables are calculated, depending on the temperature and are written to the `record` `dryAir`.

For humid air the fluid properties are calculated in the same way, but as a mixture of two ideal gases, air and water vapor.

The equations, which must be considered in a descriptive physical-mathematical model, for example in the flow resistance model, are the following:

```
...
OFFM.Media.HumidAir.HumidAirProp
HumidAir(T=T,x=
  if(includeHumidAir) then
  x
  else 0,p=p_upstream);
OFFM.Media.HumidAir.HumidAirRec
  humidair=HumidAir.HumidAirRec;
OFFM.Media.DryAir.DryAirProp
  DryAir(T=T);
OFFM.Media.DryAir.DryAirRec dryair=
  DryAir.DryAirRec;
...
```

For the inputs of the `block` class the upstream values are used.

The variable `includeHumidAir` is Boolean and can be set to true or false, depending on whether dry or humid air is called for. The fluid properties are called up from the corresponding record by means of an `if`-expression, as shown in the following:

```
cp = if(includeHumidAir) then
      (humidAir.cp) else (dryAir.cp);
```

Additionally for the calculation of humid air the equations for the water content of saturated air and for the pressure of saturated water vapor in air are relevant (Antoine Equation). The pressure of saturated water vapor in air is described by two different equations, one for the vaporization and the other for sublimation. At a temperature of $\vartheta=0.01^{\circ}\text{C}$ [3] solid, liquid, and vapor phases of water coexist in equilibrium and a discontinuity occurs between these two curves. To establish a continuous function for the pressure of saturated water vapor in air the curve for vaporization is extended up to a temperature of $\vartheta=0.0045^{\circ}\text{C}$ and the curve for sublimation is only valid up to this point.

The following approaches are used for improvement and to avoid numerical problems:

1. The Horner Scheme is applied to the polynomial for the fluid properties calculation. This has the advantage that in the converted representation no exponentials are used, only multiplications and summations.
2. `if`-expression is used in preference to `if`-clause for conditional statements.

4 Data Management

The data is divided into internal parameters and user inputs, as shown in figure 5.

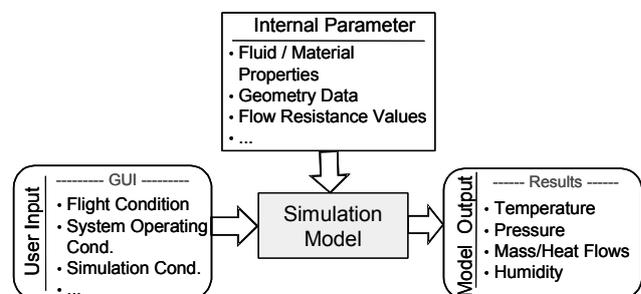


Figure 5: Input and Output Parameters

The first section in this chapter will discuss the data management of the internal parameters and is followed by the user inputs. The model outputs are not further discussed.

Internal Parameters: The internal parameters are classified into parameters, which are valid in general or only for a specific aircraft type. General parameters are e.g. the fluid and material properties and are structured in the library in the package General (see section 2.3). Specific aircraft type data are for example geometry data and flow resistance values and are contained in the package Aircraft Specific Data. All these data are stored in a record.

In addition the aircraft environmental parameters, such as ambient pressure, temperature, humidity or the external heat transfer coefficient, as well as the boundary temperatures, such as the skin temperature or the temperature of the unpressurized compartments, belong to the internal parameters. All of them are functions and mostly depend on the flight condition, for example on Mach number and altitude.

The calculation of the aircraft skin temperature depending on the absorbed solar radiation and hence on the location of the fuselage surface is described here. There are six skin temperatures, three for the upper fuselage (crown, cabin left side and right side) and three for the lower fuselage (bilge, triangle area left side and right side) whereby the calculation distinguishes between ground and flight condition. Depending on the angle β of solar radiation the absorbed radiative heat on the aircraft skin is a function of the polar coordinate φ . The direct sunbeams irradiate one half of the skin surface of the fuselage, the other half is irradiated by diffuse sunbeams. The diffuse radiation can be neglected. In figure 6 the absorbed radiative heat by solar radiation collected per projected surface area is shown. This radiative energy curve is a cosine function, which is at its maximum where it meets the irradiation angle and its minimum at the point where the solar radiation contacts the aircraft skin tangential. In case of a cloudy day solar radiation energy is attenuated by clouds. This is taken into consideration by the factor ω , below altitudes of 22000ft. The average radiative energy \bar{G} for each area can be depicted (see figure 6) and determined by following equations:

$$\dot{Q}_{\text{rad,energy}} = \omega \cdot G_0 \cdot A_{\text{proj}} \quad (4.1)$$

$$\text{with } G_0 = f(\text{altitude}), \quad (4.2)$$

$$A_{\text{proj}} = \frac{1}{\Delta\Phi} \int_{\varphi_1}^{\varphi_2} r(\varphi) d\varphi \cdot A_s. \quad (4.3)$$

The surface area, which can be calculated for the aircraft skin, is $A_s = \frac{1}{2} \cdot D \cdot L \cdot \Delta\Phi$, with $\Delta\Phi = \varphi_2 - \varphi_1$. For $r(\varphi)$ two different functions are defined for the shady (1) and sunny (2) side of the fuselage, depending on the polar coordinate and irradiation angle.

$$(1) \text{ For } \varphi > (\beta+90^\circ) \text{ and } \varphi < (\beta-90^\circ) \quad r(\varphi) = 0.$$

$$(2) \text{ For } (\beta-90^\circ) \leq \varphi \leq (\beta+90^\circ) \quad r(\varphi) = \cos(\beta-\varphi).$$

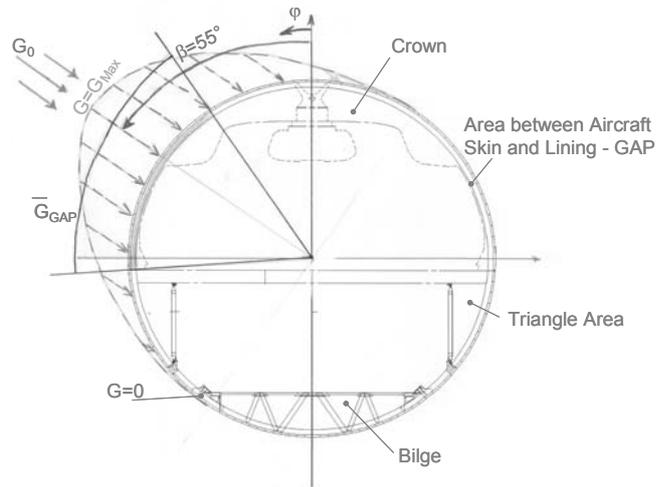


Figure 6: Aircraft Skin Temperature

The implementation in Modelica is difficult, because the integral is not varying with time but with the polar coordinate φ . So, for each range the integral is solved manually depending on the angle of solar radiation.

User Inputs: The parameter input of the user takes place in the parameter dialog, as shown in figure 7. It is contained in the aircraft template. The content of the parameter dialog is stored in a record. The statements in Modelica are set by dialog-annotations and can be found in [4].

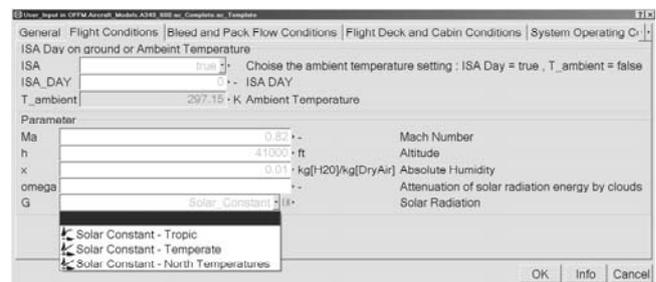


Figure 7: Parameter Dialog for User Inputs

User inputs are all operating conditions such as flight conditions, environmental control conditions and the simulation conditions, where the user has the possibility to choose whether the simulation shall run under isothermal or non-isothermal conditions and with or without the consideration of humidity.

In the parameter dialog the user has different possibilities to make his inputs. For example for the tab Flight Conditions, the user can choose, whether he wants to specify the ISA day or the temperature as an input. Depending on the setting of the parameter ISA, the corresponding input field is enabled.

It is also possible to enter a value directly in a field. To avoid errors, these input values are checked for example against minimum and maximum values.

Another possibility consists in the selection of an input in the context menu from a set of predefined values or replaceable models. For this a pull down menu is available. Figure 7 shows replaceable models for the solar constant, which is a function of altitude and depends on the different climatic zones. The user can choose whether the calculation shall run with a solar constant for tropic, temperate or for north temperatures.

The default values are predefined values for a cruise flight under normal operation conditions, with the maximum number of passengers and non-isothermal simulation conditions.

5 System Implementation

The different aircraft systems as mentioned in chapter 2.1 are implemented in the aircraft template. It is arranged in such a way that only those system modules are visible, where the user is required to give inputs. As an example the air supply system is chosen. Figure 8 depicts the air supply system for the A340-500/600, which consists of the mixer and the cabin air supply calibration module.

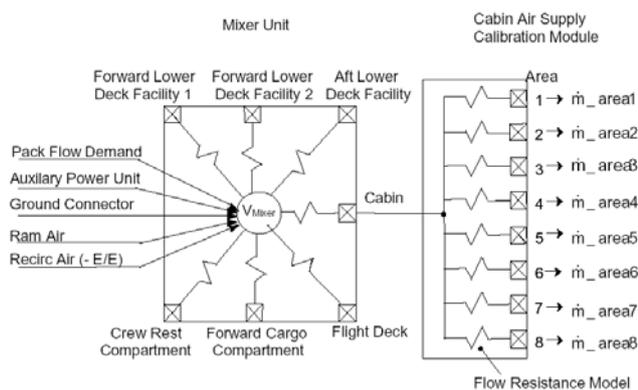


Figure 8: Air Supply System

The recirculation mass flow and the outside supply mass flow [depending on the source - ram air, APU or Engine (Pack Flow)] are input mass flows to determine over the flow resistance values the mass flows into the cabin, flight deck and forward/aft cargo compartments. The flow resistance values can

be set in the parameter dialog (tab Environmental Control System).

The air supply system supplies the air to the cabin along the whole length of each zone by four air outlets per two frame bays. On the right and left hand side of the aircraft fuselage one ceiling air outlet and one lateral air outlet position are defined, see figure 9. The ceiling outlets are not affected by any installations in the cabin; however the number and position of the lateral outlets depends on the cabin interior layout.

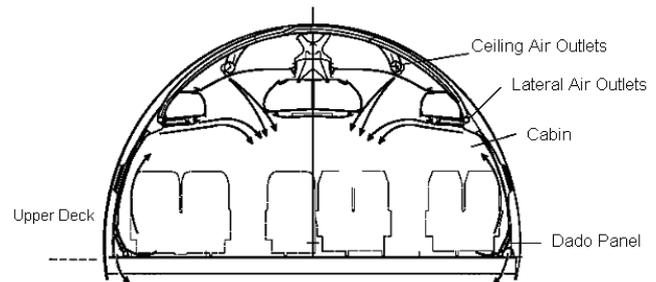


Figure 9: General Ventilation in the Cabin - Cross Section Overview

Depending on a customer layout the simulation model will be established by dragging and dropping the layout specific modules from the library into the aircraft template. The installations in the forward/aft cargo areas must be connected to the mixer unit. If a port of the mixer unit is unconnected e.g. the Lower Deck Facility (LDF) port, this means that the LDF is not installed in the cargo compartment by the user.

If a port of the mixer module is not connected the mass flow in the flow resistance model of the mixer unit shall be considered to be zero. To achieve a zero mass flow the variables at the not connected port of the mixer unit must have the same values as the variables at the port from the mixer control volume.

The modeling of such a condition in Modelica reveals to be difficult, and a complex solution has been chosen as presented hereafter.

The `DeadEndPipe`, see figure 10, represents a pressure source with the following Modelica code.

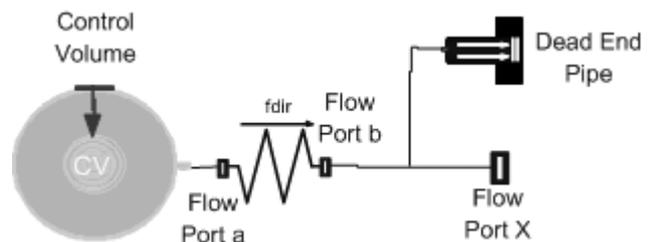


Figure 10: Unconnected Port

```

model DeadEndPipe
  extends OFFM.Icons.Flow.DeadEndPipe;
  import SI = Modelica.SIunits;
  SI.Pressure p;
  SI.Temperature T;
  SI.MassFraction x;
  parameter Boolean asSource=false;
equation
  if asSource then
    a.T = T;    a.p = p;    a.x = x;
  else
    a.H_dot =0;    a.m_dot =0;
    a.m_dot_x=0;
  end if;
end DeadEndPipe;

```

The number of connections of the flow port X is checked in the DeadEndPipe model by the following class parameter statement: `DeadEndPipe1 (asSource=cardinality(PortX==1))`. If there is no additional connection at the flow port X, the temperature T, pressure p and the water content x of the DeadEndPipe model are set to the values from the flow port a of the flow resistance model. From there they will be transmitted to the flow port b of the flow resistance model.

The cabin area is established with the specific cabin layout modules, such as galleys, lavatories, seat rows and special crew rest compartments. The individual supply mass flows for each cabin area is assigned by variables and not by a graphical connection. The reasons for that are the high number of connections, that have to be done by the end-user and the calculation of the mass flow supplied to each seat row as part of the total supplied airflow for one area. Each seat row module contains a specific number of air outlets, which varies from layout to layout, so the total number of air outlets in one area is depending on the number of installed seat rows within this area. The interface from the air supply system to the cabin is shown in figure 11.

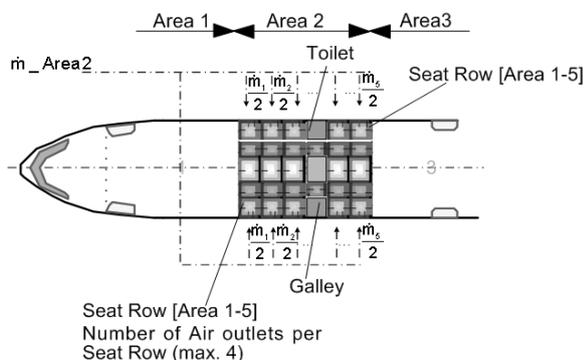


Figure 11: Interface to Cabin (I)

The implementation in Modelica proves to be quite complicated. For calculation of the total supply mass flow to an area, the number of seat rows in this area

need to be considered. In a first step the number of air outlets is not considered so that the supply mass flow per seat row is assumed to be constant.

The determined total supply mass flow for each area is written in an array, in the air supply system model, as follows:

```

inner Real m_dot_supply[8]=
  {m_dot[area1],
   m_dot[area2], ..., m_dot[area8]};

```

Then the mass flow per seat row can be calculated as follows:

```

model SeatRow
  outer Real m_dot_supply [8];
  outer Integer no_SeatRow[8];
  parameter Integer area;
  Real m_dot;
equation
  m_dot=m_dot_supply[area]/
    no_SeatRow[area],
end SeatRow;

```

The mass flow \dot{m}_{dot} must be divided and forwarded to the two internal mass flow sources in one seat row.

The number of seat rows per area is currently an user input parameter in the main parameter dialog. After establishing the layout, the user must manually count the seat rows per area and the resulting number must be set in the parameter dialog. This can easily lead to the error that the entered numbers and the numbers of inserted seat rows are not equal. At the moment there is no possibility known how to count models with the same attribute, e.g. the seat rows in one area.

In order to identify the area in which a seat row is positioned, the seat row module receives the attribute Area, e.g. a seat row in area 2: Area=2. This parameter must be entered by the user in a parameter dialog, which appears by double-clicking on the component SeatRow. An improvement for the user is the graphical position in the diagram layer, defined by a component, which can be used to set parameters. So the parameter Area could be set automatically to one, two etc. if the seat row is in a defined position range on the diagram layer.

If in the second step additionally the number of air outlets shall be considered it is necessary to enter the total number of air outlets per area in the parameter dialog and additionally the numbers of the individual air outlets in each seat row (double clicking on the component SeatRow).

The mass flow per air outlet for all seat rows in one area then can be calculated in the supply system model:

```
m_dot_perAirOutlet=m_dot_supply[area]/
no_SeatRow[area]/no_AirOutlets;
```

The mass flow for each seat row can then be determined by the multiplication of the mass flow per air outlet and the number of air outlets of each seat row.

Up to now it is insufficiently solved, how to forward the parameter `no_AirOutlets` from the Seat Row to the air supply system. The problem is that the parameter must be accessed by dot-notation – `ComponentName.Parameter`. The automatically given component name, by dragging it into the model, is not known, alternatively if the user enters a given component name and the programmer links all, errors may occur when translating the model, if a component name of a seat row is used, where no seat row is installed, for example if a toilet or galley is inserted instead of a seat row. An ideal solution would be if Modelica language provides the possibility to access parameters or variables by a wild card as “class name.*.parameter/variable”.

Currently the found solution consists in additional modules which are implemented into the aircraft template - in this way the name of the new module is known - where the number of air outlets and the calculation of the mass flow for each seat row will be done. For the user this results in an additional expense of establishing connections. The additional modules are shown in figure 12.

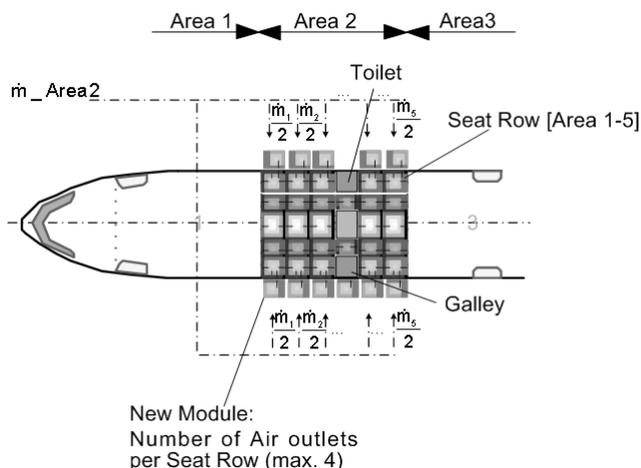


Figure 12: Interface to Cabin (II)

6 Model Application

6.1 Typical Application Procedure

The package Aircraft Model is the basis for the user. It contains for each aircraft type a specific library and the corresponding aircraft template, see figure 13. For the user only the layout specific components and the aircraft templates are of interest. For investigation of a new aircraft layout the user can select the template of the desired aircraft type out of the library.

The layout specific components need to be implemented into the aircraft template by the user, corresponding to a specific customer layout. For each dropped layout specific module in the cabin area the user has to set the following parameters: Area, in which the module is located, number of passengers in the module and the number of air outlets. The layout dependent connections to the standard fuselage modules, the connections between the layout specific modules and the connections to the system modules need to be established.

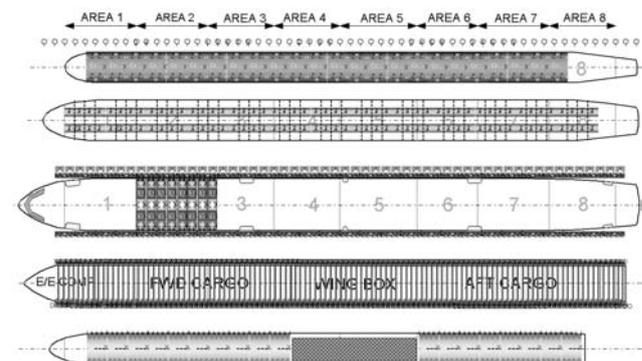


Figure 13: Aircraft Template with Cabin Layout Modules

Afterwards the main parameter dialog has to be filled out and the system configuration has to be entered and checked. Therefore the user has to work very carefully and systematically so that no inputs will be forgotten.

If this procedure has been done the user can check the model and run a simulation. The post processing is done in Matlab.

6.2 Improvement Potential

The capability of the tool Dymola/Modelica to simulate even complex systems is given. In section 4 and 5 some examples for improvement are already given.

In general there are the following suggestions for improvements based on the experience gained by developing this model.

1) Read Only

In this application case a large number of packages and sub-packages in the library shall be protected against changes by the users. They shall not be hidden in order to allow transparency and completeness, so ideally it would be possible to declare parts of the code as 'read only'.

2) Auto Connect

Another point is the number of connections that have to be done by the end-user. For each seat row the user has to make several connections. The possibility of auto connecting would make sense. An idea was to insert a new variable `AutoConnect` and the demand to specify each seat row with a predefined name. All connection statements are now written in the source code, inside an `if`-clause, which shall be valid if the `AutoConnect` is set to true. This approach is only possible if all seat rows are inserted, for which the `connect` statements are defined. Even if in one case instead of a seat row, a toilet/galley combination is inserted, and although the `AutoConnect` is set to false, when translating the model an error message appears. The reason for that is, that the compiler recognizes the `connect` statement in the `if`-clause.

3) Check-Possibility

Finally the need for a check-possibility shall be pointed out. There are several cases, where the user is required to give input values in the parameter dialog e.g. for the calculation of the supplied mass flow the parameter `total no_ofPassengers` is needed. Also when dragging a seat row in the aircraft template, for each seat row the number of passengers to consider the heat dissipation. There is at the moment no possibility to check at the end, if the values of total number of passengers (parameter dialog) is identical to the sum of passenger declared for each seat row by the user.

6.3 Model Numerical Characteristics

Several models of the aircraft fuselage with a different level of discretization have been established, to run simulations using the Dassl-Solver. The least coarse level of discretization without occurring numerical problems was the discretization of the cabin in its eight areas.

The first approach for a more detailed discretization was a cross section of the fuselage over two seat rows. Depending on the simulation time this results to numerical oscillations or it is not possible to reach steady state results. One of the reasons is that the system is stiff, due to the large difference between time constants.

Since the Dassl-Solver seems not to be able to solve such a system, first attempts were made with the RadauIIa-Solver out of Godess (Generic Ordinary Differential Equation Solver System), which is leading to an improvement.

7 Conclusions & Future Work

A short overview of the aircraft systems and the concept of the model and its library are given in this paper.

The emphasis of this paper is to demonstrate the challenges of providing a complex, user-friendly simulation model and potential solutions.

The additional requirement of providing a template to an end-user for his applications causes problems as described in section 6.2. The difficulty appears due to the fact, that the programmer does not know, which kind of library component the end-user needs at which place in the provided template and that the activities of the end-user shall be reduced to a minimum.

The development of the library and the models is an on-going process. It is planned to extend the library to consider effects for advanced modeling, such as the consideration of a human body with environment dependent heat release rather than as a constant heat source or the consideration of radiation from surfaces.

One of the next steps of modeling is to establish the complete aircraft based on a customer layout. Up to now there are several components and partial fuselage models, which have been verified. The whole development of the aircraft simulation model is subject of the validation and verification process.

References

- [1] Tummescheit H., Eborn J: ThermoFluid Modelica Library <http://www.control.lth.se/hubertus/~ThermoFluid>, 2001
- [2] Verein Deutscher Ingenieure: VDI Wärmeatlas. VDI-Verlag, Düsseldorf, 7th Edition, 1994
- [3] Baehr, H.D.: Thermodynamik. 11th Edition, Springer Verlag, Berlin, 2004
- [4] Modelica Language Specification, Version 2.1, <http://www.modelica.org>