Maschinenbau- und Betriebsinformatik mbi

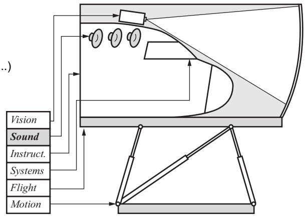
Forschungsplattform **FLUGSIMULATION**

FLIGHT SIMULATION FRAMEWORK

BUILDING BLOCKS

In order to understand the basic design requirements for a flight simulation sound system it is necessary to understand the basic building blocks of a flight simulator:

- Input/Output Devices
  - Cockpit Instruments
  - Controls (Thrust Lever, Flight Controls, Pedals,..)
- Aircraft-Systems Representation
- Flight Dynamics
- Motion Platform
- **Sound System**
- Vision System
- Instructor Station



DESIGN Principles

Discretization:
The sound simulated is considered to be the sum of various discrete sources that create sounds based on their internal state or operation.

Superposition:
The discrete sources are combined by use of superposition to generate the resulting sound. Effects generated by interference of the individual sounds have been evaluated and considered being neglectable.

Local Sound Creation:
Sound creation in the discrete sources (like aircraft system components) depends on their sound generating interaction with the environment. This interaction is determined locally by internal mechanics and functionalities and can be described with a set of relevant sound parameters. These sound parameters can efficiently only be generated in the discrete sources themselves – not outside.

Central Sound Rendering:
Contrary to sound generation, sound rendering can only be done in a central module that combines all sounds from all the sources and renders them for 3D output by taking into account position of the individual sources and sound transmission functions with respect to the cockpit.

Modularization:
All systems modeled in the simulator are seen as individual modules having a clearly defined functionality and interface to the rest of the simulation system. Real aircraft systems can be exchanged provided that they have the same functionality (e.g. it is possible to change the engines). In order to provide that same possibility in a flight simulator it is necessary to store all module specific data – mechanical as well as sound data needed for rendering – in the module context.

SOURCES AND TYPES OF SOUNDS

Discrete Sounds (event driven):
Sounds that are generated by discrete system events and can not be interrupted by the simulation are called *"discrete sounds"*. They can incorporate the sound of a sequence of sub-events and may last for several seconds. An example for a discrete sound would be the (big) bang the landing gear makes at touch down.

Continuous Sounds (permanent):
Sounds that are generated by continuous processes and that can be changed/interrupted during the simulation are called *"continuous sounds"*. Continuous sounds represent the corresponding sound to the current system state. Air-current noise is a good example for this type: Depending on the aircraft's altitude, airspeed, angle of attack, flaps and landing gear configuration and other environmental conditions, the airframe will continuously generate noise that varies in intensity and spectral characteristic.

All other sounds, even of complicated nature can be realized using these two building blocks.

SOUND FRAMEWORK

REQUIREMENTS

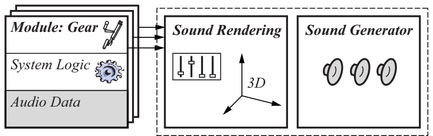
The basic requirement is to
- generate a realistic sound impression
- in real time.

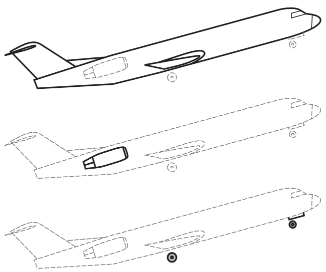
When used in flight simulation for pilot training purposes it is especially important to model
- sounds that are being created by operation and
- failure of the various aircraft system components.

Another requirement is to generate a realistic 3D sound impression in a rather small cockpit replica
The most important requirement however is
- full modularity to allow for
  - easy configuration,
  - exchange, and
  - testing of system components.

CORE COMPONENTS

The core elements of the sound system are:
- ***Modules*** representing system components with system logic, mechanical and audio data.
- ***Sound Rendering Engine*** which has two closely related functions:
  - Generating the individual sounds of all system components by using their defined sound parameters and data.
  - Creation of a virtual three dimensional sound field by placing the generated sounds on their position with respect to the cockpit. In our case we assume that all our system components have a constant position during the entire simulation. This simplifies the rendering algorithm since they can pre-compute structure damping and other correction factors.
- ***Sound Generator*** produces the individual sound signals for the built in speaker system in the simulator cockpit taking into account the speaker configuration and special hardware setting.





Sound sources in an aircraft: airframe, engine, landing gear.

SOUND SOURCE EXAMPLE: Extension of Landing Gear

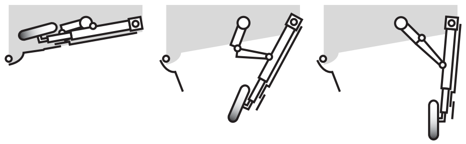
Approaching an airport for landing the pilot pulls the lever for lowering the landing gear, thus triggering the following sequence of events:
- the latches in the hull open,
- a hydraulic motor lowers the landing gear.

The resulting air resistance creates additional
- air-current noise depending on the degree the gear is extended.

If the landing gear has reached the final position,
- the mechanics locks with a possibly loud rumble so that the gear is secured in place

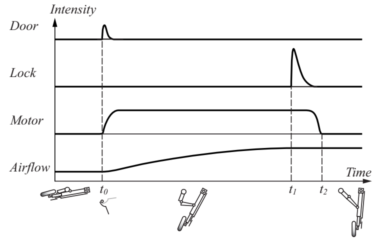
We model the corresponding sound sequence with two continuous and two discrete sounds. Continuous sounds are:
- *Air-current noise* which is (simplified for this example) determined by the parameters airspeed and the landing gear extension parameter (0-100%)
- *Hydraulic motor noise,* the characteristic of which depends solely on the motor speed (0-100% of max. rpm) for both, repeating sound patterns – similar to a starter engine in an automobile – and fundamental frequency. Let us assume it takes the motor about 5 seconds to lower the gear fully.

The following two events are modeled as discrete sounds:
- *Latch opening* creates a non interruptible characteristic noise that lasts about 0,5 seconds.
- *Gear locking* causes a fairly loud and short rumble.



Example extension of landing gear

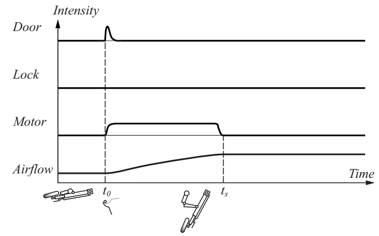
SOUND SOURCE EXAMPLE: Landing Gear – regular operation



SOUND SOURCE EXAMPLE: Landing Gear gets stuck

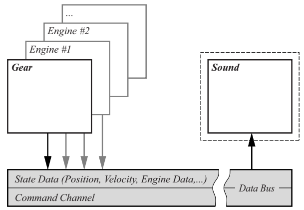
Let us assume the following mechanical landing gear extension failure:
- the hydraulic motor spins up to 50%,
- stays there for a second and
- gets stuck.

As a result the landing gear will be extended to about 60%. What would be the effect on our sound simulation?

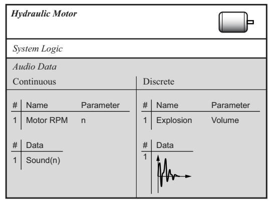


INTEGRATION IN THE FLIGHT SIMULATION FRAMEWORK

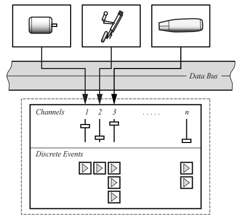
In order to integrate the sound simulation system seamlessly into the entire flight simulation framework we have chosen to use the same data bus that is common to all other modules. By design this data bus ensures a reliable and fast communication and satisfies the requirements for real time simulation.
Once initialized, the Sound Rendering Engine gets all information necessary to create the continuous sounds by listening on the data bus for the respective sound parameters. Discrete events require in addition a special trigger message containing event number and optional parameters.



Interaction between Modules and Sound Rendering Engine.



Sound Data in System Modules



Principle of sound generation within the flight simulator framework

CONCLUSIONS AND FURTHER WORK

As a proof of concept we have applied the framework laid out in this paper to designing a sound system prototype for a Full Flight Simulator for a particular airplane as a joint project with industry. Despite some technical and time constraints we were able to realize the core components and functionality of our framework.

The sound prototype was tested with airline pilots – some flying that very airplane on a daily basis and got high acceptance ratings. It turned out the additional effort in sound modularization and discretization leads to a sophisticated, detailed sound simulation, which is a major advantage in comparison to state of the art sound systems for flight training devices.

We will investigate further possibilities in modularization of this concept in the future and try to applying it to other areas like automotive.

From: *A new Functional Framework for a Sound System for Realtime Flight Simulation*, S. Vössner, R. Braunstingl, H. Ploner-Bernard, A. Sontacchi, DAFX 2005

SOUNDSYSTEM Framework