Modeling Smoke Transport in Aircraft Cargo Compartments

Jill Suo-Anttila
Fire Science and Technology Department
Sandia National Laboratories
Albuquerque, NM

David Blake
Fire Safety Section
FAA Technical Center

International Aircraft Systems Fire Protection Working Group
(for systems website posting)
Sandia National Laboratories Team Members

• Experimental
  – David Blake, Walt Gill, and Jill Suo-Anttila

• Model Development
  – Jim Nelsen and Stefan Domino

• Graphical User Interface and Code Development
  – Carlos Gallegos

• Technical Support
  – Louis Gritzo, manager of the Fire Science and Technology Department
**Goal:** Develop a CFD-based simulation tool to predict smoke transport in cargo compartments

- Improve the certification process
  - Identify optimum smoke detector locations
  - Specify sensor alarm levels
  - Identify most challenging fire locations
  - Reduce the number of flight tests

- Fast running
- Suitable for non-expert users
- Experimental data for source term characterization from FAA experiments
- Validated using FAA full-scale experiments
Software Design

Graphical User Interface

Pre-Processor

Analysis Module

Post-Processor

\[ \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j} (\rho u_j) = S_m^p \]
Pre-Processor Overview

• Provide models for different aircraft
  • Boeing 707, 727, 747, etc.
  • User defined

• Capabilities
  • Refine mesh
  • Enter fire(s) location and type
  • Enter ventilation velocities and locations
  • Enter compartment temperature and pressure
  • Add obstacles and recessed areas

• Instantaneous visual feedback
Running a Simulation
Compartment and Mesh Specification

- Execute the Pre-Processor
- Select the type of compartment
  - 707
  - DC-10
  - User Defined
- Input the dimensions
- Enter the mesh size - # of nodes
Running a Simulation
Created 707 and DC-10 Meshes

- Automatically generated 707 mesh
- Curvature captured by mesh
- Right side of screen shows selected plane
- Automatically generated DC-10 mesh
- Internal view of compartment
Running a Simulation
Recessed Area Specification

1. Advance to selected Y-plane
2. Select desired cells
3. Perform operation using buttons
Running a Simulation
Obstacle Specification

Recessed Area
Obstacle
Running a Simulation
Ventilation and Fire Specification

1. Select cells
2. Enter type of cell (inlet, outlet, fire) – cell colored to denote type
3. Use table to enter ventilation properties
4. Fire properties in file
Running a Simulation
Mesh Refinement Specification

1. Select the plane for refinement
2. Use refinement tool
3. Enter level of refinement
Running a Simulation
Running the Analysis Code

- Analysis - - - Run Analysis
- Status monitored on screen
Smoke Transport Analysis Code

- Curvature of compartment is resolved on grid
- HRR, MLR are time varying inputs (as measured in FAA experiments)
- Species tracking: presently soot, CO, and CO$_2$ but addition of more or different species possible
- Simulation time = 1 hour per minute of real time
- Validated using FAA full-scale experiments
Post-Processor

Allow users to manipulate data in a variety of ways

- contour plots
- time history of field variables
- 3D smoke visualization in time
• Thermocouple temperature rise
  – 0 - 60 seconds
  – 0 -120 seconds
  – 0 -180 seconds
• Light transmission
  – 30 and 45 sec (ceiling and vertical)
  – 60 sec (vertical - high, mid, low)
  – 120 sec (vertical - mid and low)
  – 180 sec (vertical - mid and low)
• Gas species concentration rises
  – 0 - 60 seconds
  – 0 -120 seconds
  – 0 -180 seconds
Status of FAA Full-Scale Validation Experiments

• 707 experiments completed
  – Baseline – center fire
  – Attached – sidewall fire
  – Corner – corner fire
  – Determined leakage ventilation had no impact on data
  – All 707 experiments were conducted without ventilation

• DC-10 experiments in progress
  – Ventilation validation
  – Three fire locations
Future Activities

• Continue validation of the smoke transport code
  – Finish code modifications
  – 707 validation comparisons
  – DC-10 validation comparisons

• Release of code to small user community (Spring ’04)
  – Includes theory and users manual

• Revisions and final release of code (Feb ’05)