

Commercial Aircraft-Cabin Egress: The Current State of Simulation Model Development and the Need for Future Research

M. C. Court

The University of Oklahoma
School of Industrial Engineering
Norman, Oklahoma, USA
E-mail: mcourt@ou.edu

There has been increasing interest in developing simulation models capable of analyzing commercial aircraft-cabin egress under both non-life-threatening and life-threatening scenarios. At issue is the ability to accurately simulate human behavior within non-toxic environments, as well as the debilitating effects that toxic environments (e.g., fire and smoke) have on human-decision making. A set of criteria has been identified by the Federal Aviation Administration for developing simulation models capable of analyzing commercial aircraft-cabin egress. These criteria are used to (a) compare the capabilities and limitations of four aircraft-evacuation models in existence today, (b) identify the issues that need to be addressed when developing these types of models, and (c) propose a new paradigm for developing aircraft-cabin egress models.

Keywords: Aircraft-cabin egress, aircrafts, certification test, evacuation, human behavior, FAA regulations

1. Introduction

Manufacturers of new passenger (commercial) aircraft, and certain modified-passenger aircraft (e.g., an increased number of rows to the original cabin design) are required to show compliance with 14 Code of Federal Regulations, Part 25, Section 803, *Emergency Evacuation*. This requirement is frequently referred to as *The 90 Second Rule*. The manufacturer must show that with half of the available exits blocked, a full load of passengers and crew (flight attendants, pilots, etc.) can safely evacuate the aircraft into a darkened hangar in 90 seconds or less. If the manufacturer's evacuation test is successful, the aircraft is deemed *airworthy*. If the manufacturer is not successful, the test must be run again until airworthiness is achieved, with each test requiring an entirely new set of passengers and crew. This requirement provides a performance-based *certification test* of the emergency-evacuation system of an aircraft.

Certification testing has proven to be quite stressful and costly to the manufacturer. Today's certification test costs an average of \$2.3 million, involves over 4,000 people, and requires three years of planning [1]. The costs rise exponentially if the manufacturer is not successful on the first certification test (recall, the test

must be conducted until airworthiness is achieved and each test requires a new set of passengers and crew).

Adding to the monetary cost of every certification test is the risk of injuries to the human-test subjects (i.e., the passengers and crew). Thus, the requirement to use children during certification tests was eliminated by the Federal Aviation Administration (FAA) in 1970. However, passengers subject to the highest risk of injury have been women over the age of 50 [1]. For example, in October 1991, a 65-year-old woman tripped at the aircraft exit of an MD-11. After falling headfirst down an escape slide, she suffered a broken neck and was left paralyzed [2]. Due to this and other injuries, the FAA allowed modifications to The 90 Second Rule. The manufacturer (McDonnell Douglas Aircraft) was allowed to (a) use ramps rather than slides, and (b) increase exterior illumination. When the manufacturer used the ramps and increased exterior illumination, there was a time penalty of 28 seconds imposed on the second certification test. So rather than having 90 seconds to complete the evacuation, the evacuation had to be completed within 62 seconds. The penalty was based on observing past certification test films where about (a) 15 seconds was taken by the crew to inflate the slide, and (b) 13 seconds in total hesitation time was realized at the slide (i.e., passengers hesitated before jumping down a slide, but did not hesitate when running onto the ramp) [2].

Consequently, there has been increasing pressure to improve certification tests' safety, even if the resulting tests give up some realism [3, 4]. Nevertheless, there is one realism that certification tests have never incorporated: the dynamic environment of an aircraft cabin during an accident (i.e., fire and smoke). So today's certification tests do not provide the manufacturer, the airline industry, or the authorities (FAA) a scientific means for predicting the evacuation performance of passengers, crew and aircraft under various parameters (e.g., number of disabled passengers, number and location of flight attendants, number and types of exits, fire, and smoke).

2. Criteria for Developing Simulated Egress Models

Since requiring manufacturers to perform certification tests under actual accident conditions is ethically unacceptable, the need then exists to develop evacuation models capable of simulating (a) various cabin configurations, (b) the dynamic environment of fire, and (c) passenger behavior. While the physical architecture of the cabin and the spread of fire and smoke can be modeled easily with today's existing technology, at issue is the ability to accurately predict human behavior. More specifically, the issue is the ability to simulate the physical and psychological effects fire and smoke have on human behavior and decision making.

The following was presented by Court and Marcus [5] on behalf of the FAA, at the 1996 International

Advisory Group for Aerospace Research and Development (AGARD) Symposium on Aircraft Fire and Safety, as the criteria for developing a simulated egress model for studying aircraft-cabin evacuations:

1. The model must be capable of analyzing various aircraft-cabin configurations without requiring changes to its source code. This will give various end-users (manufacturers, the FAA and airline carriers) the capability of analyzing an aircraft's emergency system without requiring them to be versed in a specific programming language (or the model's logic) [4].
2. The model must run in real time or near real time. This will support the ability to analyze the evacuation procedures that are followed, and the impact those procedures have on evacuation time and passenger-survival rates.
3. The model must be able to conduct simulations of both certification tests and accident evacuations.
4. The model must consider relationships among passengers. For instance, the impact on the evacuation behavior of a parent traveling with an infant, versus a passenger traveling alone, must be incorporated.
5. The model must consider the impact a flight attendant's behavior has on passengers. This feature will allow *passenger management* to be explored, such as determining the optimal number of flight attendants per passenger load, or the optimal location for each flight attendant, particularly for when queues build in front of exits. Such a tool is non-existent today. For example, the standard-evacuation procedure is to have flight attendants stationed directly at over-wing exits. However, it has been demonstrated that placing them at the end of the row (where the over-wing exit is located) allows the flight attendant the capability of redirecting passengers to other exits (which have no queue), and thus reduces evacuation times [4].
6. The model must offer dynamic behavior as opposed to behavior that is fixed at the time of model execution. That is, the model must allow the behavioral characteristics of the passengers to change over time. For instance, when passengers are exposed to (or spend more time in) a life-threatening environment, they become more willing to take risks (e.g., jump over seats) [4].
7. The model must take into account the dynamic, toxic environment of fire and consider the physical as well as psychological effect of fire and smoke on human behavior.
8. The model must support simulation output analysis, designs of experiments, and sensitivity analysis. Since the goal of the manufacturer is to perform certification testing only once, data for statistical analysis is non-existent. And, fortunately, commercial aircraft accidents are rare events. In addition,

most survivors are either (a) reluctant to be interviewed, or (b) recall events out of sequence (or with an inaccurate timetable) [4]. Thus, there is no tool (or data) in existence that supports the goal of identifying the most important factors influencing passenger survival.

9. The model must provide animation of the evacuations to support model validation and presentations.
10. The model should operate on a PC platform. This will ensure the model is portable.
11. The model must have a high degree of user friendliness, again, so that the model can be used by a wide set of end-users.

Notice that if this set of criteria is followed, (a) the testing on evacuation systems of commercial aircraft is moved into a more realistic, operational environment (involving smoke, fire and the threat to human lives), with no risk to human subjects; (b) manufacturers will have access to a tool that is capable of evaluating conceptual designs of various aircraft structures before the production and certification testing phases; and (c) airline carriers will be able to optimize emergency

procedures and personnel requirements by aircraft-cabin type (model).

3. Commercial Aircraft-Cabin Egress Models

There are a limited number of models available for conducting analysis on simulated aircraft-emergency egress. In fact, most evacuation models of commercial aircraft are not released to the public. The impediments for releasing these models are the regulations and the constraints that the United States or other governments could impose on the model developers and end-users [6].

The following presents a literature survey on the four aircraft-evacuation models in use today, their assumptions, capabilities and limitations. I studied the capabilities and limitations of these existing aircraft-evacuation models over a three-year period (under an FAA grant). The four models reviewed are the Gourary Associates (GA) model, AIREVAC, EXODUS and ARCEVAC. Since each model was funded by different organizations, each has varied capabilities and limitations.

Table 1 displays an overview of the four models as based on the set of criteria established for aircraft-cabin

Table 1. Desired features of commercial aircraft evacuation models

Features		GA	AIREVAC	ARCEVAC	EXODUS	Desired
Legend						
• Existing Capability + Planned Capability						
1	Analyze various cabin configurations without changes to the source code			•	•	•
2	Run in real time or near real time	•		•	•	•
3	Perform Certification Test evacuations	•	•	•	•	•
4	Fire	•		+	•	•
5	Smoke			+	•	•
6	Toxic Environments			+	•	•
7	Perform accident reconstruction			+	•	•
8	Passenger behavior		•	•	•	•
9	Crew behavior			•	•	•
10	Supports passenger management studies			•	•	•
11	Supports bonding or passenger relationships			•		•
12	Simulates dynamic passenger and crew behavior			•	•	•
13	Physical effects of fire on passenger behavior			+	•	•
14	Physical effects of smoke on passenger behavior			+	•	•
15	Physical effects of toxicity on passenger behavior			+	•	•
16	Psychological effects of fire on passenger behavior			+	•	•
17	Psychological effects of smoke on passenger behavior			+	•	•
18	Psychological effects of toxicity on passenger behavior			+	•	•
19	Supports statistical analysis (repeated, independent runs)	•		•	•	•
20	Supports experimental designs					
21	Animation	•	•	•	•	•
22	PC platform	•		•		•
23	User-friendliness	Medium	Low	High	Low	High
24	Amount of knowledge required by the user	Low	High	Low	High	Low

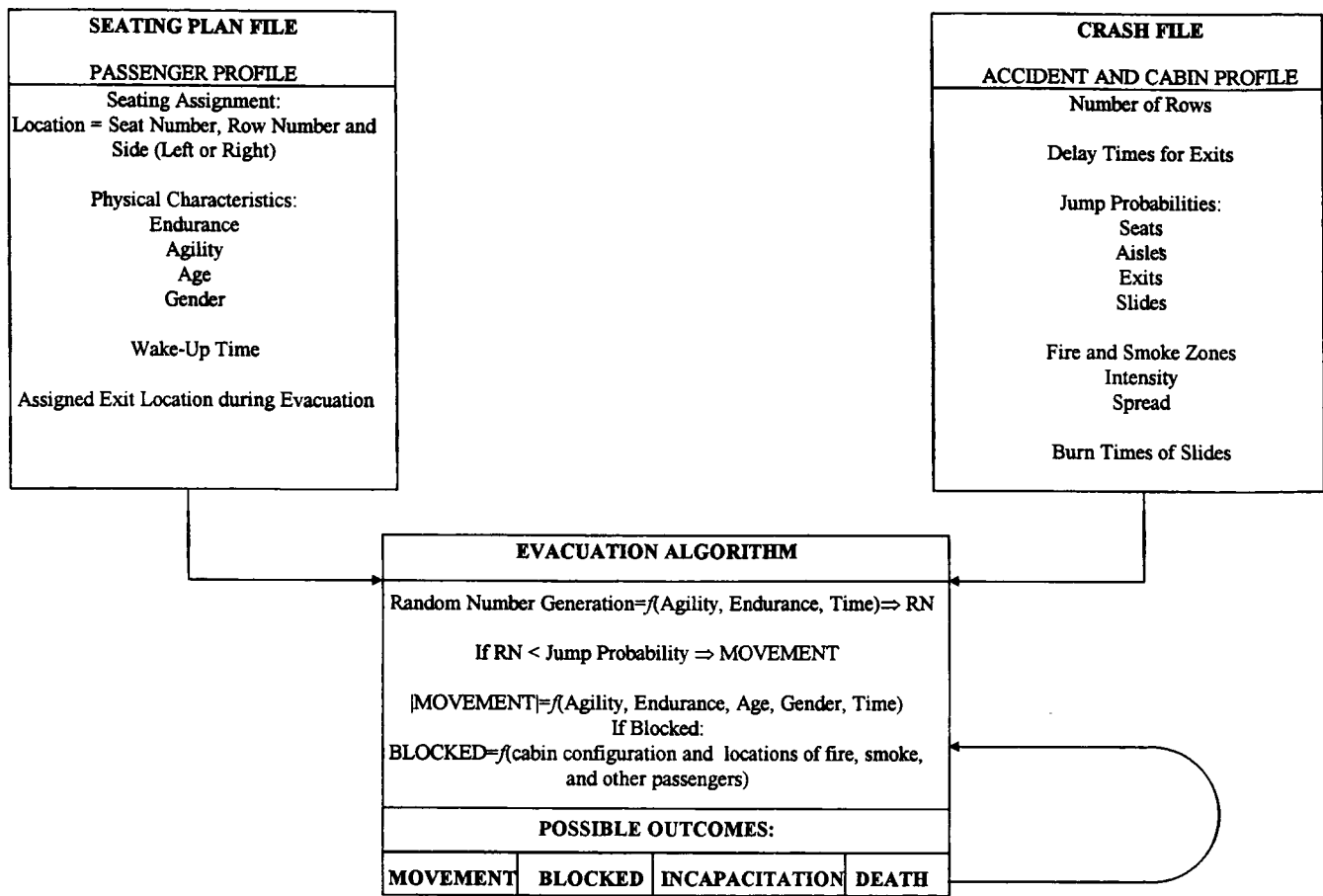


Figure 1. Gourary Associates (GA) model

egress models. The GA model was obtained and extensively studied for two years. The other models either are not released to the public due to the funding institution's request, or are considered to be still in the development stage. Those models were studied via the model developer's supplied documentation of the model.

3.1 Gourary Associates (GA) Model

The Gourary Associates (GA) model, Version 6.02 [7, 8], is capable of simulating passenger egress from single-aisle aircraft under both certification and accident (fire and smoke) conditions. The software's development was initially funded by the Department of Transportation through a *Small Business Innovative Research* contract. The GA model runs on any personal computer, produces a 2-D (top-down) view of the aircraft-cabin interior and passengers, and runs at near real time. An overview of the model, its files and interfaces are shown in Figure 1. The software is written in Microsoft BASIC, where the user enters input factors on the (a) aircraft structure and its environment, and (b) passenger data.

The GA model allows the user to control (a) the number of rows within the aircraft, (b) the amount of time to delay the opening of exits, (c) the movement and speed of passengers through exits and within the

aircraft, (d) fire and smoke zones, inclusive of their spread and intensity, and (e) the burn time of slides.

The user describes the passengers in terms of their seating assignment (location), endurance, agility, age and gender via the seating plan file. The user can also control the reaction time of passengers through the *wake-up time* assignment. Specifically, the wake-up time is the amount of delay a passenger encounters when trying to unbuckle his seat belt and initiate movement. During accident investigations, it is common to find deceased passengers still in their seats with their seat belts buckled [4]. Consequently, a large wake-up time assignment for a certain percentage of the passengers supports accident-reconstruction analysis.

The seating plan file also contains the exit a passenger is assigned to use during the evacuation, and thus, generates the path for that passenger. The path generated is based on the shortest feasible distance to their pre-assigned evacuation exit. Passenger movement towards their pre-assigned evacuation exits is achieved by generating a set of random numbers from an algorithm that uses the agility and endurance of the passenger as input. However, passenger movement can be impeded based on various blockages (dead or incapacitated passengers, inoperable exits, etc.).

Passengers maintain their initial endurance and agility levels in simulations not involving fire (certification evacuations). If the simulation involves fire or smoke, the endurance value of the passenger is continually reduced, based on the time the passenger is exposed to the hazardous environment. This reduction causes the probability of movement to decrease. If the value of the endurance drops below a certain threshold value (a global variable within the model), the passenger is considered *incapacitated*. If the endurance of the passenger falls further, the passenger is considered *dead*, and could block other passengers' movement.

The GA model's three greatest assets are (a) the platform, (b) the low level of computer knowledge required by the user, and (c) the ability to perform multiple runs. However, the GA model has the following drawbacks that prevent it from being a versatile tool for simulating aircraft egress:

1. The program can simulate only single-aisle aircraft configurations, with a maximum of three seats per row, on each side of the aisle. Hence, wide-body, dual-aisle aircraft cabins cannot be simulated using this model.
2. The architecture of the interior cabin is extremely limited, even within the family of single-aisle aircraft. That is, the configuration is limited to six exits (i.e., one front, one back, and four over-wing

exits), where over-wing exits must be located at the midway point of the cabin. For example, to simulate an MD-80 aircraft cabin consisting of 38 rows where the over-wing exits are located at rows 23 and 24, 12 *fake* rows (non-occupied seats) must be placed in the crash file for the over-wing exits to be located at the correct row location [9].

3. The algorithm for movement is global in its approach and does not allow individual reactions to external factors.
4. There is no capability of simulating human behavior, only human movement.
5. In addition, there is no means to distinguish between the crew and passengers. Therefore, simulating passenger and crew interaction is not possible.

3.2 AIREVAC

AIREVAC, developed by Schroeder and Tuttle [10], and sponsored by the Air Transport Association of America, is specifically designed for B-727 aircraft certification tests. The model cannot be used to simulate any other aircraft configuration without reprogramming the source code in Simscript. The original objective for its development was to investigate the impact disabled passengers have on evacuation times. During its development, Schroeder extended its applicability

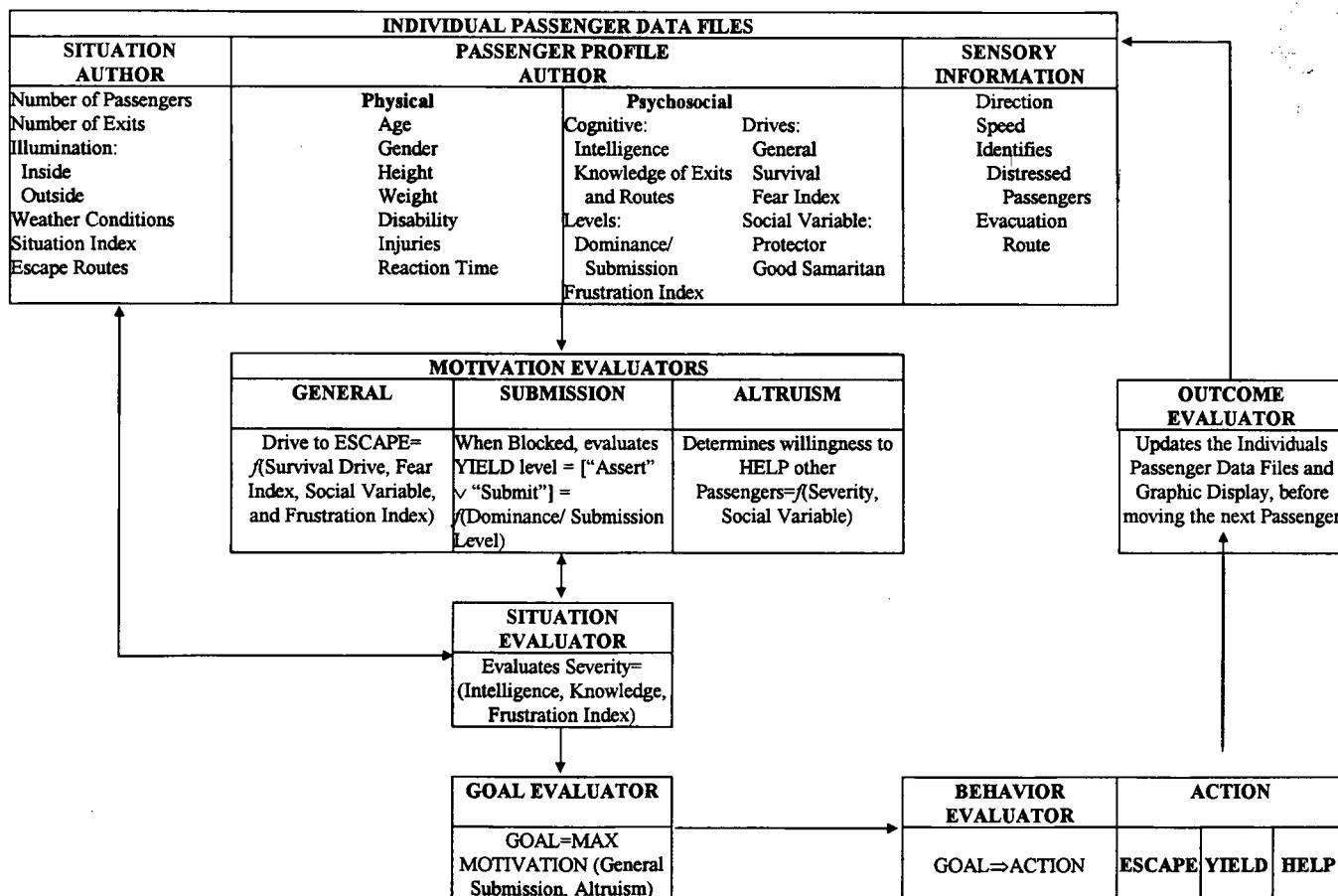


Figure 2. AIREVAC model

into a wider scope, a *psycho-social model*, in that it has the capability of simulating passenger behavior during evacuations.

The program uses many of the parameters commonly used by aircraft-cabin evacuation models, such as the number of passengers, the location of each passenger, and the speed of movement (reaction time). In addition, however, the program uses a separate, extensive set of psycho-social parameters to describe the reaction of a passenger during the evacuation process. Some of these parameters are the frustration index, the dominance and submission level of the passenger, and the passenger's knowledge of exits and routes.

The AIREVAC model has a set of functional units called *evaluators* (refer to Figure 2). Each of these evaluators applies rule-based algorithms to determine passenger movement during the evacuation process, and communicates the outcome to other evaluators. The various functional units of the model are the *situation evaluator*, the *motivation evaluators* (*general*, *submission* and *altruism*), the *goal evaluator*, the *behavior evaluator*, and the *outcome evaluator*. Two *authors* are used to establish the settings for the physical environment of the evacuation (*situation author*) and the passenger's physical and psychosocial characteristics (*passenger profile author*). Physical characteristics are age, gender, height, weight, disabilities and injuries. Psychosocial characteristics are (a) cognitive variables, such as general intelligence and knowledge of aircraft exits and routes, (b) levels (dominance and submission), (c) drives, such as the fear index and survival drive, and (d) social variables, such as protector or Samaritan. The following is a description of these evaluators [10]:

1. The *situation evaluator* uses the data from the *situation author* to establish the psychological and physical conditions during the evacuation, and generates a situation and frustration index for each passenger. The evaluator also establishes a hierarchy of escape routes (paths) for each passenger.
2. The *general motivation evaluator* determines the net motivational drive to escape. This is a pooled sum of the variables representing the survival drive, the fear index, the social variable (protector or Samaritan), and the frustration index.
3. The *submission motivation evaluator* program is only called when a passenger encounters a *block* (e.g., a passenger is competing with another passenger for the right-of-way) in his movement and determines whether an individual will *assert* himself or *submit* (yield).
4. The *altruism motivation evaluator* determines an individual's tendency to help other passengers.
5. The *goal evaluator* determines which of the three primary motivations, *general*, *submission*, or *altruism*, is the strongest. It then selects one of three goals for the passenger: *Escape*, *Yield*, or *Help*, respectively.

6. The *behavior evaluator* transforms the current strongest goal (from the goal evaluator) into behavioral action. For *Escape*, movement is continued on the current path. For *Yield*, the passenger will stop his movement and allow other passengers to occupy his path. For *Help*, the passenger will assist the distressed passenger who is blocking his path. The behavior evaluator then determines a specific behavioral outcome, or action (e.g., passenger unfastens his seat belt, the passenger moves to the aisle, the passenger pushes other passengers, the passenger panics, the passenger falls).
7. The *outcome evaluator* executes the action requested by the behavior evaluator, and updates the passenger data files and graphic display.

Although AIREVAC allows the behavior and decision making process of human beings to be simulated in great detail, the program has the following shortcomings:

1. The program can simulate only one type of cabin arrangement at a time without reprogramming. To reprogram, the user must have thorough knowledge of the programming language Simscript and the AIREVAC logic.
2. The program does not run at real time or near real time. On the average, it takes several hours to run a 90-second evacuation test [3].
3. There is currently no means to simulate the toxic environment of smoke and fire.
4. The model cannot perform multiple, independent simulation runs of the same evacuation scenario.

3.3 ARCEVAC

ARCEVAC is currently being developed by the Aviation Research Corporation on behalf of the Transportation Development Center in Canada. It is a PC-based, emergency-evacuation model that allows behavior of all human occupants (passengers and crew) to be simulated during egress [11].

The model operates within a "windows-type" environment, where the input data is entered through a series of graphical user-interface windows. The ARCEVAC model has been designed around a variable-parameters logic, where the user can opt to have the algorithms run via supplied data, or the users can supply their own data to the algorithms. The advantage of this model will lie in its ability to model a large set of egress scenarios—from simple evacuations (where behavioral aspects of the passengers and crew, and hazardous environments are not considered), to complex evacuation scenarios (where the physiological and behavioral aspects of the passengers and crew, and hazardous environments are considered). The more complex scenarios will be established by having the user provide detailed input parameters to the model's algorithms (e.g., heat index of the fire, location

and smoke spread). That is, the user will be able to “enrich” the model to the desired level of complexity.

ARCEVAC uses relative scales for quantifying the passenger profile (distribution) through a *Passenger Selection Specification* window. That is, the user supplies data to a *passenger description tree*, consisting of three branches. The first branch contains the percentage of male versus female passengers, while the second branch divides the male and female passengers into two percentage categories: those below the age of 40 and those above the age of 40. The last branch divides the age categories based on percentages with disabilities (through a separate window interface). The passenger description tree is further defined by providing the passenger types: those traveling for pleasure and those traveling for business. If passengers are traveling together, they are then identified via a *group definition specification*, indicating the group number, the group’s size and group type (business, friend, or family).

Individual passengers receive their physical parameters (weight, height, cross-sectional area, agility and ability to resist hazardous environments (endurance)), and psychological parameters (group and individual selfishness index), by assigning each category of the passenger description a *distribution*. For example, the percentage of men below age 40 is assigned a distribution for their heights. The distribution is used to obtain the individual passenger’s height value. All physical distributions follow a truncated-normal distribution where the user must enter the minimum, maximum, mean and standard deviation parameters. The psychological characteristics are based on relative scales (percentages). The *selfishness* and *group selfishness* indices indicate the tendency of the passenger to be selfish towards strangers (passengers not belonging to the same group), and selfish towards those passengers belonging to the same group, respectively.

A similar mechanism, the *Crew Selection Specification* window, is used for defining crew types and their behavior. However, the crew’s psychological profile does not include the selfish indices. Instead, crew behavior is controlled through its *mission type*. The mission type indicates how the crew will assist passengers in evacuating the aircraft (checking the aisles, opening doors, etc.). The *valor* factor indicates how long the crew will attempt their mission (assist passengers), while the *Crew Performance Table* indicates how well the crew accomplishes their mission. The Crew Performance Table is a 6×6 matrix containing six *crew performance categories*: excellent, good, fair, poor, none, N/A (no crew is simulated, just passenger evacuation). Once the crew member is assigned a performance category, his ability to assist passengers is defined by a set of six values: Range, P(Deterred), P(Depart), Check Interval, Agility Modifier, and Exit Door Access Time. For each crewmember’s performance category, these values are defined as:

1. The effective distance the crew member’s voice can be heard—Range.
2. The probability the crew member will deter (change the behavior or direction of) a passenger—P(Deterred).
3. The probability the crew member can remove a passenger from a blocked exit queue—P(Depart).
4. The amount of time lag before the crew member will initiate his mission—Check Interval.
5. The effect on passenger agility—Agility Modifier. For example, if the crew member is categorized as excellent, the Agility Modifier is 25%. So, if a passenger responds to that crew member’s command (e.g., moves away from a blocked exit), the passenger’s agility level will be increased by a factor of 1.25.
6. Exit Door Access Time is the time it takes the crew member to open an unblocked, unjammed exit.

During model execution, the mission assignment for the crew is controlled via three modes: no mission assigned, fixed mission assigned with no run-time control, or fixed mission assigned with run-time control. If no missions are assigned, the model’s logic will invoke the crew to (in the following prioritized order): (a) guide passengers through exits, (b) assist immobilized passengers, and (c) evacuate the crew. If fixed missions are assigned, the user’s defined mission for the crew will be executed at the simulated time, as specified by the user. If run-time control is invoked by the user, the run-time mission commands will take priority over the mission command compiled at model execution. In other words, the user can change the behavioral aspects of the crew during the model’s run time.

The *Human Behavior Specification* window allows the user to specify the percentage of passenger and crew members who will respond in a Calm, Alert, Nervous, Panicky, or Shock manner, by defining a cumulative-percentage value for each of those manners, over the range [0, 99]. *Behavior Modifiers* are events within the evacuation (e.g., loss of illumination, flooding, or explosions) that can be triggered to impact the behavioral responses of the passengers and crew. For example, for a given percentage of the passengers, Calm may be assigned the range 0 to 40; Alert, 41 to 63; Nervous, 64-80; Panicky, 81-90; and Shock, 91-99. Thus, 41% of the passengers will remain calm, 23% will be alert, 17% will be nervous, 10% will panic, and 9% will be in shock when the interior-cabin illumination is reduced to zero.

The software incorporates graphics and animation for the motion of the passengers and the conditions of the aircraft cabin during the evacuation simulation. The program is written in Borland C and C++ with object-oriented features, such as generic-class libraries. However, the present version of the model can be

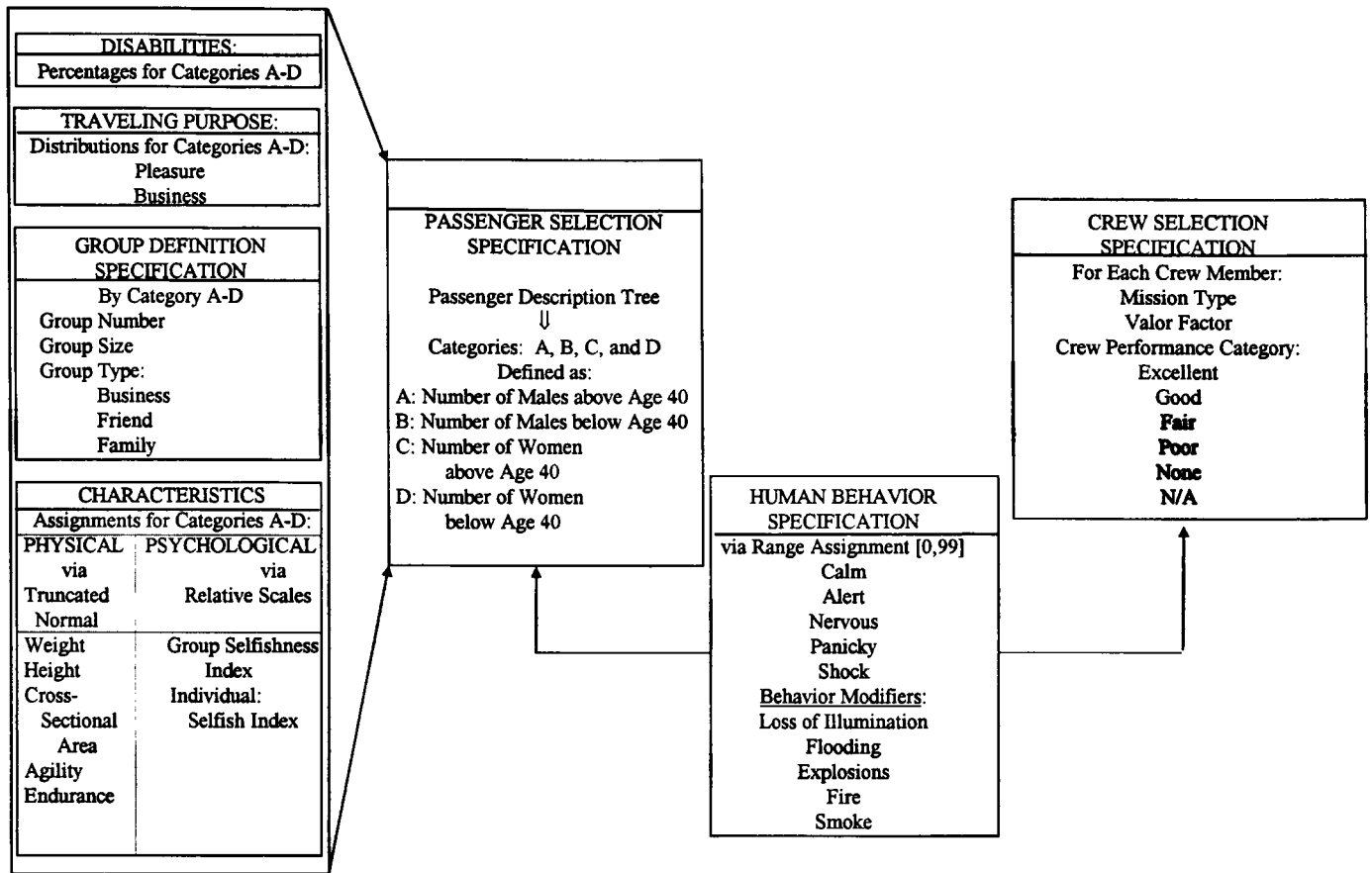


Figure 3. ARCEVAC window interfaces

used only for certification tests, while future versions are expected to be used for simulating hazardous conditions (fire, smoke and toxicity). The current advantages of ARCEVAC are:

1. The high level of user friendliness and the lack of knowledge required by the user to run the model. That is, the user does not have to be versed in the underlying simulation language.
2. The PC platform and the ability to run in real time.
3. The ability to simulate both human and crew behavior so that passenger management issues can be explored.
4. The ability to link passengers to groups so that their behavior responses are dependent on their internal and external group classifications.
5. The ability to perform multiple simulation runs.

The major drawback to the ARCEVAC model is that truncated normal distributions are used to assign the physical and psychological characteristics of the passengers, and thus the user cannot assign features on an individual basis. In other words, ARCEVAC takes a global approach to the assignment of individual physical attributes, as well as individual behavior. Therefore, ARCEVAC does not support experimental designs. That is, you cannot place the same set of passengers into various aircraft and treat their physical or psychological characteristics as a controllable factor.

3.4 air-EXODUS

EXODUS is a suite of evacuation models currently being developed at the University of Greenwich. In its infancy, EXODUS was used to simulate the evacuation of large populations of individuals from mass transit vehicles (e.g., trains and ships) and from buildings (e.g., cinemas and lecture halls) [12]. The original program was written in a programming environment called G2 and ran on a SUN SPARC-1 station. Since 1993, EXODUS has evolved into two distinct packages: building-EXODUS and air-EXODUS [13]. The former is devoted to analyzing structural evacuations, while the latter is specifically tailored for aircraft evacuations. Both are on a PC platform and written in C++. air-EXODUS is capable of simulating the interactions of passengers, crew, fire and aircraft structures.

The behavior and movement of humans are driven by a set of global and local heuristics and rules, as categorized by five submodels: *Movement*, *Behavior*, *Occupant*, *Toxicity*, and *Hazard*. The submodels' operations are defined over a region of space within the *Geometry* submodel. An overview of the submodels is shown in Figure 4 and described below [12, 13]:

1. The Geometry submodel contains the physical description of the aircraft's interior, inclusive of seats, aisles, windows, and exits. The interior is represented by a layered network of nodes, where each node contains the information about its single

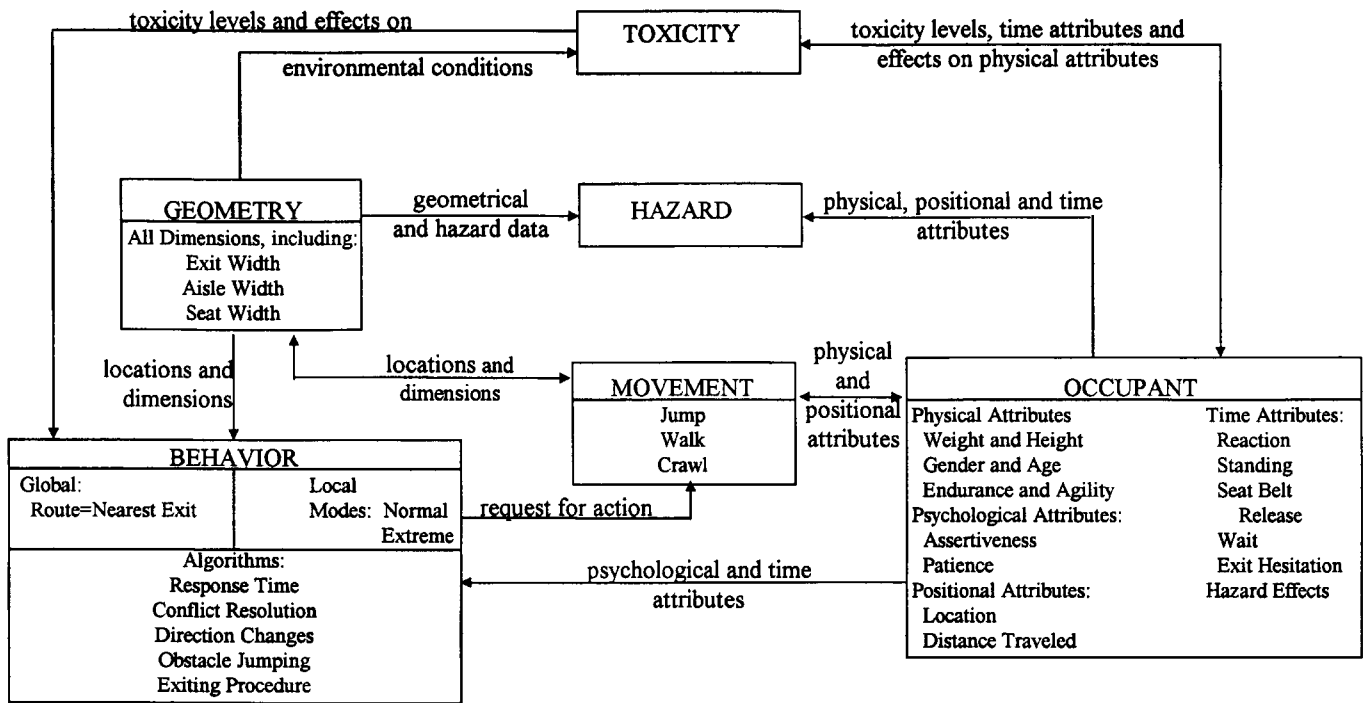


Figure 4. air-EXODUS submodel interaction (Galea, et al. [13], modified)

occupant (e.g., seat), and if pertinent, the environmental conditions (temperature, carbon dioxide, oxygen levels, etc.). Nodes are spaced about 0.5 meter apart throughout the interior of the cabin, and are connected by a set of arcs. These arcs maintain feasible paths from one node to the next. In other words, a passenger will occupy a set of nodes and is not allowed to occupy (or move to) a set of nodes representing an interior wall.

2. The Movement submodel controls the physical movement (walking, crawling and jumping) of the passengers from one set of positional nodes to the next. It is responsible for determining whether the passenger has the capability to perform the requested physical action (via the Behavior submodel), by interfacing with the Occupant submodel (that contains the physical attributes of the passenger). For example, the submodel ensures that the passenger has the appropriate agility level to jump over a seat before initiating that movement.
3. The Behavior submodel determines the response of a passenger to the prevailing situations, based on two rule-levels: *global* and *local*. Global rules generate an overall escape plan for each passenger based on the nearest exit, or the exit he has been directed to by a crew member. Local rules allow the individual's personal behavior attributes to operate in two modes: *normal* or *extreme*. In the normal mode, a passenger will adhere to the global behavior and will respond immediately to crew member's orders. In the extreme mode, passengers are allowed to ignore global behavior (e.g., move towards an exit that is not the nearest exit). The Behavior submodel

passes the resulting decision (request for action) to the Movement submodel.

4. The Occupant submodel describes five sets of characteristics (attributes) for each passenger: physical (age, weight, height, gender, endurance, and agility), time (reaction, seat-belt release, waiting, and exit hesitation), hazard effects, positional (location and distance traveled), and psychological (patience and assertiveness). Some of these attributes are fixed throughout the simulation, and some of the factors change as the result of input(s) from other submodels. For example, if during the simulation a hazardous environment is invoked (e.g., smoke), passengers will have their waiting time attributes reduced. That is, passengers are more likely to tolerate longer queues during a non-life-threatening evacuation than during a life-threatening evacuation.
5. The Toxicity submodel is only activated when the user specifies environmental conditions simulating fire. Fire causes a variety of toxins (CO_2 , CO, HCN, etc.) to be released into the cabin, since much of the cabin is constructed of synthetic materials. Consequently, during accident investigations it is common to find that passengers are overcome by these toxins before the fire reaches their locations [3]. The Toxicity submodel determines the effects these toxins have on passengers, and communicates this information to the Behavioral model, which in turn passes it on to the Movement model. For example, the Toxicity submodel determines the effect smoke has on the passenger's ability to travel, by identifying a reduction in the passenger's walking speed,

or by signaling to the Movement module that the passenger must now crawl.

6. For simulations involving fire, the Hazard submodel controls the interior environmental conditions (e.g., oxygen levels and heat intensity) through a set of nodes located at two heights (in relation to the passenger): head height and knee height. Note that this submodel does not generate the initial fire conditions, but relies on the user to enter the data manually. This module also controls the timing of the opening and closing of all exits.

The decision analysis performed within the Behavior submodel is the crux of air-EXODUS. There are five algorithms responsible for decision analysis within the submodel: *response time*, *conflict resolution*, *direction changes*, *obstacle jumping*, and *exiting procedure*. The algorithms receive their data values on (a) each passenger from the Occupant submodel, (b) the aircraft's interior architecture from the Geometry submodel, and (c) the aircraft's interior-atmospheric conditions from the Toxicity submodel. Based on the algorithm invoked and the resulting decision, a specific request for action is sent to the Movement submodel. Each algorithm has its own function and is described as follows:

1. The response time algorithm determines the time a passenger consumes in getting out of his seat, as a function of three time attributes: (a) reaction time, (b) the time the passenger will take to release his seat belt, and (c) the time it takes for the passenger to stand upright.
2. The conflict resolution algorithm is called upon when two (or more) passengers wish to occupy the same set of nodes at the same time. The passenger with the highest assertiveness attribute is dominant (and, therefore, will occupy the set of nodes) over the passenger(s) with the lower assertiveness level.
3. The direction change algorithm is invoked when any of these three events occurs: (a) a crew member issues an order to the passenger that, if followed, would divert the passenger from his intended path, (b) the queue time incurred by the passenger exceeds his wait time attribute, or (c) the passenger encounters a concentration of toxins.
4. The obstacle jumping algorithm is triggered when a passenger encounters either (a) an architectural block (e.g., a seat or debris), (b) a slower passenger and his patience level is exceeded, or (c) a dead passenger.
5. The exit procedure algorithm considers the width (from the Geometry submodel), the physical dimensions of the passenger(s), and the exit hesitation (the passenger's time attribute), to determine the (a) maximum number of passengers that can pass through an exit at the same time, and (b) the time each passenger will take to pass through the exit. For example, it generally takes more time for

a passenger to escape through an over-wing exit than a floor exit (door exit). Additionally, more passengers at one time can escape through a door exit than through an over-wing exit.

air-EXODUS has many desirable features, including an extensive graphical display of the movement of each passenger during the simulation run. In addition, when simulating evacuations involving fire and smoke, the model provides detailed information on (a) the amount of toxins absorbed by each passenger, and (b) the time taken by each passenger to exit the aircraft (or the time until his/her death). The software is capable of simulating both wide-bodied, dual-aisle and narrow-bodied, single-aisle aircraft and is portable across many types of platforms (e.g., PC Windows-based and UNIX). On a 486/25 MHz machine, a simulation run takes an average of three minutes to evacuate 400 passengers [13].

The major disadvantages of air-EXODUS are:

1. The inability to simulate relationships among passengers. That is, the Behavior submodel only considers the state of the *current passenger* when invoking its algorithms, since air-EXODUS has no means to identify (or link) passengers. For example, the obstacle jumping algorithm does not consider the relationship of the slower passenger to the current passenger. What if the slower passenger is the child of the current passenger?
2. As with ARCEVAC, there is no means to use the model for experimental designs. That is, there is no means to control behavioral responses of the passengers from one replication to the next. Thus, it would be difficult to identify the most important factors influencing passenger survival between aircraft types.

4. Proposed Approach

The four aircraft-cabin evacuation models reviewed take a global approach to controlling the behavior of their test subjects. That is, only one algorithm is used for simulating the effects of external factors on the test subjects (human beings) and thus, individual responses to the same stimuli are not allowed. In addition, none of the four models takes into account the relationships among passengers.

More importantly, validation is critical to the successful use of a model as a predictive tool and involves testing to ensure that the model accurately reflects the behavior of the real system. None of the literature available on the four models addresses the issues involved when validating the simulated egress behavior.

Of the paradigms available to date, object-oriented programming (OOP) has two inherent features that may lend it to modeling human behavior and decision making during aircraft-cabin egress:

1. OOP lends itself to the modeling of complex systems by supporting modular construction.

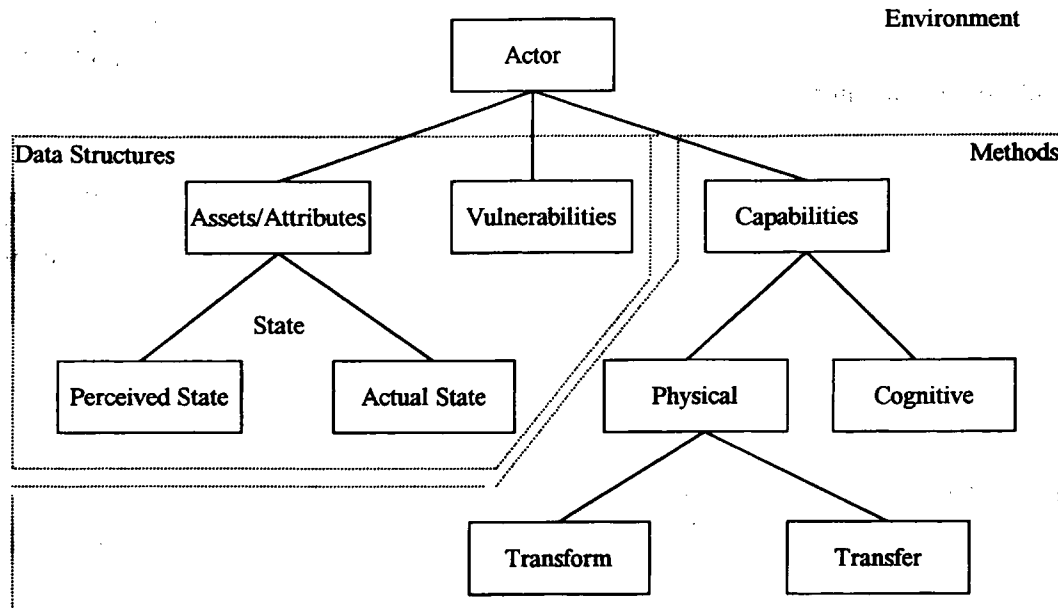


Figure 5. Actor-centered description [14]

2. OOP yields a one-to-one correspondence with the physical world and, thus, eases the burden of model validation.

In 1988, Burns and Morgeson [14] published a construct for simulating systems involving endogenous decision making that draws upon an object-oriented paradigm. Their work proposes describing the system in terms of a *suite of actor classes* (collection of object classes) whose endogenous decisions impact the performance and behavior of the system. They suggest a model where all actors, including pseudo-actors (environment), follow an actor-centered description (Figure 5). Each actor class requires data structures (assets, attributes and vulnerabilities) and methods (cognitive and physical capabilities), described as follows:

1. *Assets* are discernible characteristics and *attributes* are descriptive characteristics. The actor's own assets and attributes comprise the *actual state*, while the *perceived state* is the actor's perception of its surroundings (the environment and other actors). The *state* of the actor is the combined data structures of the actual and perceived state. This state data is the input to the *cognitive inference engine* of the actor.
2. The actor can physically move (*transfer*) or change (*transform*). Transformation takes place by modifying the actor's assets/attributes. *Vulnerabilities* represent degradation to the actor's capabilities via the reduction or destruction of the actor's assets.
3. An action space for cognitive capabilities and activities describes the decision set and state of each actor. A *cognitive event* (decision) is capable of (a) delaying decisions, (b) invoking physical activity, and (c) changing the action space. By delaying decisions and changing the action space, an actor then

has the ability to "change its mind" (non-monotonic reasoning).

However, the actor-centered description proposed by Burns and Morgeson [14] is not a pure object-oriented paradigm, since knowledge bases (production rules and heuristics) and inference engines are utilized for achieving cognitive activity. This approach would equate to developing data, knowledge, and method structures for each actor (passenger) and pseudo-actor (environment). Hence, the danger of adopting this approach is the possibility that the objective of supporting real-time simulation may not be met.

The proposed framework of Figure 6 incorporates the actor-centered description (for passengers and the crew), but avoids the use of knowledge bases and inference engines by providing a pure object-oriented paradigm. To achieve this pure object-oriented approach, a new set of class objects must be defined that contains (a) the data describing the environment that the actors need to make decisions, and (b) the applicable rules (functions) for the actors when making the decisions. In essence, the actors obtain their data and functions by copying these new objects, or parts of the new objects, into their own class definition when making decisions.

The new class objects are the *Coordinate*, *Navigate*, *Advancement*, *Path*, *Responsibility*, *Panic* and *Block* objects. These objects replace the knowledge bases and inference engines of Burns and Morgeson [14], while the *Passenger* and *Crew* objects generate the actor instances representing the passengers and crew, respectively. The environment (pseudo-actors) is represented by the *Fire*, *Smoke*, and *Toxicity* objects. The *Synchronizer* object is used to advance the simulation clock, keep track of state of the environment, and invoke the ordered movement of actors and pseudo-actors. An

overview of the construct in Figure 6 is described below:

1. The *Coordinate* object maintains the aircraft-cabin architecture and generates, through its *coordinate* functions, a *map* of the aircraft cabin for the *Passenger*, *Crew*, *Fire*, *Smoke*, and *Toxicity* objects. The map contains information on all of the exit, row, and aisle locations. The *Passenger*, *Crew*, *Fire*, *Smoke*, and *Toxicity* objects may copy the coordinate functions of the *Coordinate* object. Thus, depending on their capabilities, the *Passenger*, *Crew*, *Fire*, *Smoke*, and *Toxicity* objects are able to store and update their positions and distances from other objects.
2. The *navigate* functions of the *Navigate* object are also available for the actor objects (*Passenger* and *Crew* objects); and if applicable (for accident reconstruction scenarios), for the pseudo-actors (*Fire*, *Smoke*, and *Toxicity* objects) to copy. The *navigate* functions allow the actors and pseudo-actors the capability of choosing headings (direction) for movement.
3. The *Path* object is called upon to generate possible paths for the actors (*Passenger* and *Crew* objects) and pseudo-actors (*Fire*, *Smoke*, and *Toxicity* objects) based on their positions, headings, and cognitive abilities to access the environment. In essence, the *Path* object represents the *eyes* of the actors and pseudo-actors. Again, how far the actor is able to see depends on the defined capabilities of the actor.
4. The functions of the *Advancement* object are used to move actors and pseudo-actors to their requested positions.
5. Data from the *Block* object is used to keep actors and pseudo-actors from moving into inaccessible positions. The *Block* data types consist of architectural (seats, walls, etc.), human (passenger and

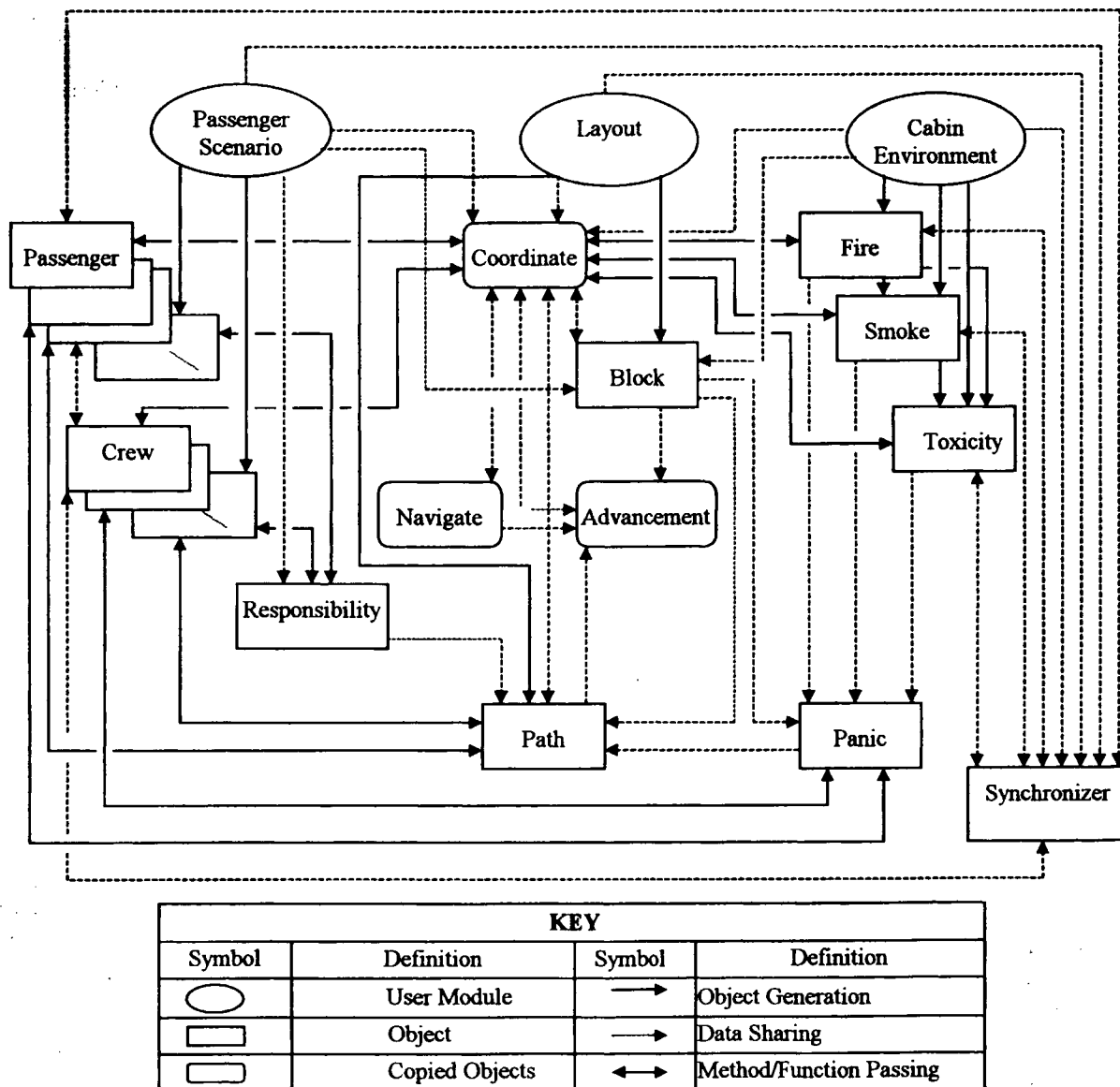


Figure 6. Proposed object-oriented paradigm

crew), and environmental (fire, smoke, and toxicity) obstacles.

6. The *Panic* object directly influences the actor's ability to reason and react. The *Panic* object is used to degrade the actor's assets/attributes and thus represents vulnerability. Degradation is realized by the inability to call on other objects (send messages or copy), such as the *Advancement* object. Again, the call to the *Panic* object will depend on the actual and perceived state of the actor, and thus is a function of the actor's capabilities.
7. The *Responsibility* object is used to bind objects together. This is one of the vehicles used to establish a psychological profile for each actor, as well as a means to distinguish flight attendants and crew members from passengers. A flight attendant is expected to assist and direct passengers during an evacuation. In this paradigm, a flight-attendant actor has access to the internal data of other objects. This is achieved in object-oriented programming by designating objects as *friends* to other objects. Friend-type objects also include passengers traveling together. The amount of internal data sharing depends on the relationship type and the amount of responsibility an actor has toward another actor.

Note, how 'capable' the actor is at using the *Coordinate*, *Navigate*, *Path* and *Advancement* objects depends on its physical and cognitive capability objects. For example, the possible paths a passenger can construct, and how many times a new path is generated, depend not only on how often the *Path* object is called upon, but is also a function of the actor's actual and perceived states. Thus, path generation is a function of the type of evacuation being performed (certification or accident reconstruction) and the actor's (a) immediate environment (fire, smoke and toxicity levels), (b) ability to access its current path and blockages, and (c) time spent in hazardous environments.

The construct supports pre-defined biological hierarchies but allows distinction between objects within the same biological class. For example, although females have many similar physical characteristics, they do not have the same physical capabilities; likewise, they do not have the same cognitive reasoning abilities. Thus, a distinction based on technical knowledge can be made between a female passenger and a female flight attendant. That distinction is incorporated through the ability to copy the *Coordinate*, *Navigate*, and *Advancement* objects. The flight attendant is expected to have knowledge of the aircraft's configuration and therefore has more access to the functions and data of the aforementioned objects than an average passenger does. Also, consider a female passenger traveling alone versus one traveling with an infant. The mother is bound to her child and, therefore, would be expected to ensure that the child is evacu-

ated safely. In this construct, when the mother actor is generated, a copy of the bonding function from the *Responsibility* object is copied into the *mother actor*. The mother actor is now tied to her *child actor* where the child's state is input to the cognitive object of the mother actor.

The construct supports the objectives for developing aircraft-cabin egress models by supporting (a) a varied passenger and crew profile, (b) a wide variety of aircraft cabin configurations, and (c) the capability of simulating various hazardous environmental conditions.

The construct is currently being implemented at the University of Oklahoma's School of Industrial Engineering. The study is at the validation stage, with simulations being validated against actual certification test data. The study is also identifying the physical and psychological parameters most influential to passenger survival. A preliminary set of parameters is currently being incorporated into the model, as are the environment objects of fire, smoke and toxicity.

5. Conclusions and Future Research

Of the four models reviewed, ARCEVAC and air-EXODUS have the greatest potential for meeting the criteria set by the FAA for simulating emergency egress. However, none of the models reviewed supports the execution of experimental designs. At issue is the ability to identify the most important factors of passenger survival. The proposed model is capable of conducting experimental designs since physical, as well as behavioral, characteristics can be treated as controllable factors.

Consequently, the proposed approach is the only model aimed at analyzing the capability of an emergency system to accommodate a wide variety of passenger types (including both physical and behavior profiles). Hypothetical scenarios that need to be explored by the new approach are, "What if the same set of passengers were simulated to (a) behave in the same way, (b) with the same emergency situation invoked, and (c) only the aircraft cabin's interior architecture is changed?" That is, the question to be explored is, "When is behavior the deciding factor on passenger survival, or when is the design of the aircraft's interior cabin the deciding factor?" Additionally, the emergency procedures during egress need to be addressed and the number (or placement) of flight attendants within the aircraft should be analyzed for optimal scenarios.

For the model developer, however, there is an unavoidable question: to what degree (if any) of liability is he to be held accountable if an accident results in human injuries or deaths, and his model was used to assist (a) the manufacturer in designing the aircraft's emergency system, and/or (b) the airline carrier in establishing evacuation procedures, or flight attendant

manpower and location requirements? While the questions for the authorities, however, are:

1. "Are computer-based simulations an improvement to the current (performance-based) certification test?"
2. If the answer above is *yes*, then, "To what degree will (or should) computer-based simulations of an aircraft's emergency system replace (or supplement) the requirement of certification testing?"
3. And then, "What will be (if any) the model validation requirements imposed on software developers before the model can be used for replacing or supplementing the current certification tests?"

In general, the consensus is that for a more realistic test of the aircraft's emergency system to be performed, simulation must be an integral part of that system's evaluation. Therefore, future research must be aimed at developing, validating and implementing simulation models for certification and accident investigation purposes [3]. This opens a new area of research for simulation-validation techniques—validating human behavior and decision making, particularly under life-threatening scenarios. Currently the approach is to rely on accident and certification data. Recall, however, that the majority of certification tests are only performed once, and, luckily, commercial aircraft-accidents are rare events. Then how will these models be validated against samples of size one? Current research is aimed at determining what behavior is common among all types of emergency egress (e.g., building evacuations) and extending that historical database across various scenarios.

6. References

- [1] Shook, W., Senior Analyst for McDonnell Douglas Aircraft, *Technical Meeting: FAA Evacuation Model Parameter Sensitivity Study*, University of Oklahoma, School of Industrial Engineering, March 6, 1995.
- [2] Dornheim, M.A. "Evacuation Tests Without Slides Approved." *Aviation Week and Space Technology*, Vol. 61, January 1993.
- [3] Marcus, J. "A Review of Computer Evacuation Models and Their Data Needs." Department of Transportation/Federal Aviation Administration Technical Report: DOT/FAA/AM-94/11, Washington, DC, 1994.
- [4] Marcus, J.H., Manager of the Protection and Survivability Laboratory, for the Federal Aviation Administration at the Civil Aeromedical Institute, *Technical Meeting: FAA Evacuation Model Parameter Sensitivity Study*, University of Oklahoma, School of Industrial Engineering, November 10, 1994.
- [5] Court, M.C. and Marcus, J. "Use of Object Oriented Programming to Simulate Human Behavior in Emergency Evacuation of an Aircraft's Passenger Cabin." *AGARD Conference Proceedings 587: Aircraft Fire Safety*, Chapter 34, pp 1-7, 1997.
- [6] Schroeder, J.E. Technical Discussion (via Telephone): *FAA Evacuation Model Parameter Sensitivity Study*, University of Oklahoma, School of Industrial Engineering, October 1994.
- [7] Gourary, B.S. "Simulation of Evacuation from a Single-Aisle Airplane: A Comprehensive Report." Gourary Associates Inc., Montclair, NJ, 1993.
- [8] Gourary, B.S. "PC-Based Simulation of the Evacuation of Passengers from a Transport Airplane." Gourary Associates Inc., Montclair, NJ, 1994.
- [9] Jayarama, S. and Court, M.C. "Evacuation Model Parameter Sensitivity Study: Project Progress." Technical Report, School of Industrial Engineering, University of Oklahoma, January 1996.
- [10] Schroeder, J.E. and Tuttle, M.L. "Development of an Aircraft Evacuation (AIREVAC) Computer Model, Phase I: Front End Analysis and Data Collection." Technical Report: SwRI Project Number 12-4099, Southwest Research Institute, San Antonio, TX, 1991.
- [11] Aviation Research Corporation. "Development of ARCEVAC An Emergency Evacuation Simulation Model—Beta Version 1.0." Transport Canada Technical Report: rjs-92-122, Transportation Development Center, Montreal, Canada, 1994.
- [12] Galea, E.R. and Galsarsoro, J.M.P. "EXODUS: An Evacuation Model for Mass Transport Vehicles." Civil Aviation Authority Technical Report: CAA Paper 93006, London, 1993.
- [13] Galea, E. R., Owen, M. and Lawrence, P. "The Role of Evacuation Modelling in the Development of Safer Air Travel." *AGARD Conference Proceedings 587: Aircraft Fire Safety*, Chapter 36, pp 1-12, 1997.
- [14] Burns, J.R. and Morgeson, J.D. "An Object-Oriented World-View for Intelligent, Discrete, Next-Event Simulation." *Management Science*, Vol. 34, No. 12, pp 1425-1440, 1988.

7. Additional Reading

- Lewis, C. and Cherry, R.G.W. "Technical Evaluation Report: AGARD- Propulsion and Energetics Panel—88th Symposium on Aircraft Fire Safety." *AGARD Conference Proceedings 587: Aircraft Fire Safety*, T1-T19, 1997.