FORMAL ASPECTS OF PROCEDURES: THE PROBLEM OF SEQUENTIAL CORRECTNESS

Asaf Degani
San Jose State University, CA; and
NASA Ames Research Center
adegani@mail.arc.nasa.gov

Michael Heymann
Technion, Haifa, Israel; and
NASA Ames Research Center
heymann@cs.technion.ac.il

Michael Shafto
NASA Ames Research Center, Moffett Field, CA.
Mshafto@mail.arc.nasa.gov

A formal, model-based approach is proposed for the development and evaluation of the sequences of actions specified in procedures. The approach employs methodologies developed within the discipline of discrete-event and hybrid systems control. We demonstrate the proposed approach through an evaluation of a procedure for handling an irregular engine-start on board a modern commercial aircraft.

In complex human-machine systems, successful operations depend on an elaborate set of procedures provided to the human operator. These procedures specify a detailed step-by-step process for configuring the machine during normal, abnormal, and emergency situations. The adequacy of these procedures is vitally important for the safe and efficient operation of any complex system. In high-risk endeavors such as aircraft operations, maritime, space flight, nuclear power production, and military operations, it is essential that these procedures be flawless, as the price of error may be unacceptable. When operating procedures are inadequate for the task, not only will the system's overall efficiency be thwarted, but there may also be tragic human and material consequences (Degani and Wiener, 1993).

In commercial aviation, for example, crew interaction with the aircraft is specified through a set of Standard Operating Procedures (SOPs) (Federal Aviation Administration, 1995). In the event of a normal task (e.g., configuration of the aircraft before takeoff), an abnormal condition (e.g., high engine temperature on start-up), or an emergency situation (e.g., engine fire), procedures are set in place to support the crew in managing the situation. Procedures assist the crew along a path of pre-defined sequences of actions; the objective is to quickly "drive" the system to some safe, yet still efficient, configuration. It must be recognized, however, that an unpredictable constellation of circumstances including machine (e.g., component failure), human (e.g., making a mistake), and environmental factors (e.g., low ambient temperature) can interfere with operations and lead to a sub-optimal configuration (see Mosier, Palmer, and Degani, 1992, for one example).

From the organization's point of view, a procedure represents a collective agreement on the "best" way to achieve *both* safe and efficient operations (Wieringa, Moore, and Barnes, 1992). Nevertheless, there are many documented cases in which the procedures provided to the crews are not the "best" (Degani and Wiener, 1997). For example, one U.S. airline's abnormal procedure for coping with asymmetrical-flap-extension (which can have a significant effect on lateral control of the aircraft) had to be rewritten when it was found to be inaccurate. The problem? The power supply for activating the flaps following asymmetrical flap extension, was different from the standard configuration for this model aircraft. The airline that originally specified the non-standard power supply configuration failed to modify the procedure accordingly. (The inaccurate procedure was in effect for some five years before it was detected).

Based on our survey of several U.S. airlines, we have noted that the process of designing a procedure is accomplished informally. That is, a Flight Manager and/or several experienced pilots discuss and then (re)-design the procedure based on their knowledge, experience, and intuition. Once the procedure is reviewed by the regulating agency's (e.g., FAA's) inspector, the procedure is approved, accepted, and provided to all flight crews. Other industries that we surveyed, such as nuclear power, maritime, and space, use similar procedural design processes.

We believe that current procedural design processes should be augmented with an in-depth evaluation of the procedure in terms of its [1] sequential correctness, [2] ability to deal with out-of-norm configurations, [3] compatibility with the user interface, [4] vulnerability to human error, [5] capability of meeting the demands from the operational environment, and [6] consistency with other procedures and policies. In this paper we suggest an approach for describing and analyzing procedures in terms of *sequential correctness*.

APPROACH AND LANGUAGE

Procedures constitute sequential execution trees (i.e., conditional instruction sequences) of user interaction with the machine. Their aim is to guide the user in operating the machine correctly and reliably, so as to achieve well-defined task goals and specifications. It is quite clear that in order to formulate a correct and efficient operational procedure, the

procedure designer must have a clear and unambiguous understanding of the machine's behavior under all (relevant) operating conditions.

The approach proposed in the present paper is aimed at enhancing current practice by augmenting it with a formal mathematical methodology that provides a systematic method for procedure "synthesis." Two elements must be in place to perform such synthesis: [1] a formal model of the machine's behavior and [2] a formal representation of the procedure's task goals. Such a model can be based on any one of several existing or emerging modeling formalisms for (untimed) discrete-event systems or (timed) hybrid-systems (Ramadge and Wonham, 1987; Heymann, Lin and Meyer, 1997).

Our objective is to develop formal approaches for designing and evaluating procedures (see Degani and Heymann, 1999, for a similar approach for evaluating interfaces). The focus of this paper is on the sequential correctness problem. From a theoretical standpoint, we strive for an approach that describes the human-machine-environment system and its many embedded interactions in a clear (e.g., mathematical) language that allows for a detailed description, synthesis, and analysis. From a practical standpoint, we seek an approach that provides a reliable design process, e.g., such that fixing one procedural deficiency will not generate another deficiency somewhere else—a well-known and common problem in procedure development.

Language

The foundation of our approach is a formal description of the human-machine system in terms of its behavior. We use the Finite-State-Machine theory to model system behavior. The following is a brief description of two graphical representations of this theory: the State Transition Diagrams and the more modern Statecharts formalism (Harel, 1987).

In Figure 1a we have three states A, B, and D (depicted as rounded squares) and several transitions (depicted as arcs).

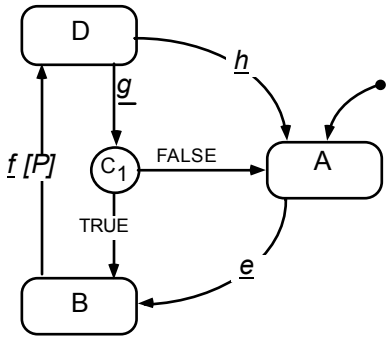

Figure 1a. *State-transition-diagram.*

The symbols *e, f, g,* and *h* stand for events that trigger transitions among the machine's states. The bracketed [P] is a condition, such that the transition from state B to D takes place when event *f* occurs and condition P is TRUE (at the same time). C_1 is also a condition such that when *g* occurs and C_1 is evaluated FALSE, the machine transitions to A; if C_1 is evaluated TRUE, the machine transitions to B.

The first Statecharts enrichment is concurrency of processes. Two related processes can be placed together in a so-called AND state, separated by a dotted line (Figure 1b). The resulting super-state S is an abstraction of the two concurrent processes X and E. Process X is made up of two sub-states Y and Z, and process E is identical to the process in Figure 1a. The question as to which sub-state is initially occupied when entering super-state S is resolved by the small default arrows (), which point to states Y and A.

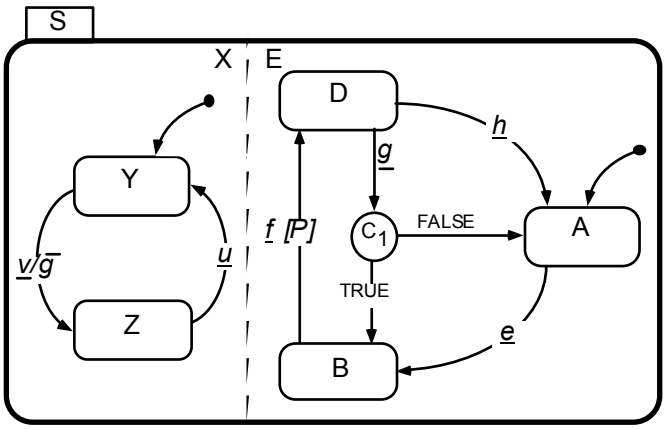

Figure 1b. *Concurrency and broadcast.*

The real subtlety with which Statecharts models concurrency is in its treatment of output events, or actions. Here the machine can generate actions to change its own configuration. Consider process X in Figure 1b: When event *v* occurs and the transition labeled *v/g* is taken, the action *g* (an output event, denoted with a hat) is immediately activated. This event is broadcast to the entire network, and perhaps causes further transitions in other processes. And indeed, in process E, action *g* will cause a transition out of state D (into A or B depending on how condition C_1 is evaluated).

The ability to arrange processes in a concurrent manner and to broadcast information among processes sums up two important features of the Statecharts language. These features of Statecharts allows us to describe the behavior of a system in a clear and concise way. Below, we will use the Statecharts language to describe one human-machine system.

EVALUATION

To illustrate our approach we evaluate an abnormal procedure used in commercial aviation. In evaluating this human-machine-environment system, we [1] describe the machine and procedure, [2] model the system, [3] define the task goals and specifications, and [4] analyze the necessary sequence of events to execute this procedure.

Machine and Procedure

Normal engine start in the Boeing B-757 aircraft follows this sequence of actions: engagement of the engine starter, opening of the fuel control switch once the engine is at the appropriate speed, and automatic cut-out of the engine starter once the engine is running on its own. In the case of abnormal start events–such as when the engine is not starting after starter engagement and application of fuel, a high engine

temperature on start-up, or pneumatic or electrical supply interruption–the pilots are instructed to follow a prescribed procedure. The procedure, the IRREGULAR ENGINE START for aircraft, specifies the sequence of immediate actions that must be performed by the crew to avoid further damage to the engine and to shut it down properly. Figure 2 is a copy of the procedure as it appears in the pilots' manual.

IMMEDIATE ACTION

FUEL CONTROL SWITCH . . . . . CUTOFF

ENGINE START SELECTOR. . . . . .GND

Motor for 30 seconds or until EGT is below 180, whichever is longer (unless no oil pressure).

NOTE

If starter cutout has occurred, reselect GND when N2 is below 20%

If problem was other raped EGT rise:

ENGINE START SELECTOR. . . . . .OFF

Figure 2. *Irregular Engine Start.*

The sequence of actions for the irregular engine start is to first close the fuel valve (fuel control switch – cutoff), and then engage the ground starter (engine start selector – GND). These two actions should be done immediately–that is, from the pilot's memory—and not by opening and reading a procedures book. Once these two steps are executed, the pilot is then instructed to engage the starter and crank ("motor") the engine for 30 seconds or until the engine's Exhaust Gas Temperature (EGT) falls below 180 degrees. This means that if 30 seconds have elapsed and the engine temperature is still higher than 180 degrees, the pilot should continue to motor the engine (with the starter engaged), until the temperature subsides. The pilot is then cautioned that engine motoring should not be continued if there is a no oil pressure, because "dry" motoring will severely damage the moving parts in the engine.

The procedure further cautions the pilot that if the ground-starter has disengaged automatically (starter cutout) as part of the normal start, the pilot should re-select ground (starter) when the speed of the second stage fan (N2) is below 20 percent. In all cases, the pilot should wait for the fan speed to drop below 20 percent before engaging the ground starter, because engaging the ground-starter when the engine fan is rotating at a high speed will damage the starter. (This is somewhat similar to engaging the starter in an automobile when the car engine is running). The careful reader immediately notes that there are some problems in the wording and arrangement of actions, conditions, and notes in this procedure. And indeed, identifying these deficiencies in a systematic way is the objective of this paper. We begin by modeling the system involved in the irregular engine start.

Model of the System

Figure 3 is a model of engine behavior, given pilot interactions, during an irregular engine start. Three concurrent processes are depicted: Engine, Fuel Control Switch, and Engine Start Selector. The initial state of the "Engine" process is IDLE (note the small arrow). The pilot starts the engine by first moving the engine start selector switch (depicted in the lower-left process of Figure 3) to GND. This event (*gnd*), in turn, broadcasts the event *ground starter* to the engine. Now the engine is motoring.

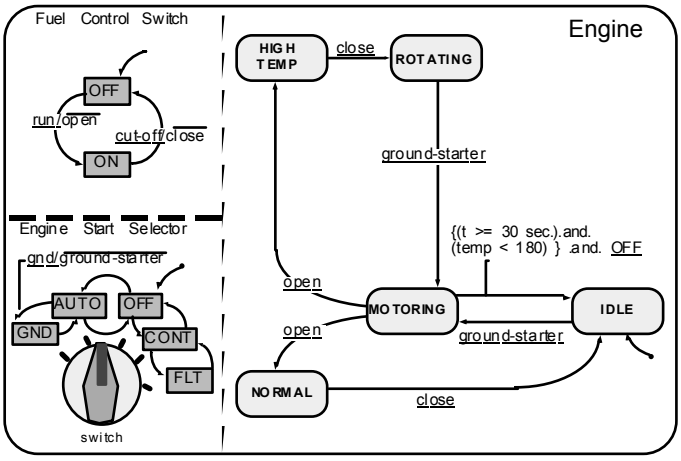

Figure 3. *Model.*

Once the engine fan (N2) reaches a speed of 25 percent, the pilot places the "Fuel Control Switch" to ON (depicted in the upper-left process in Figure 3). This transition from OFF to ON, triggers the output event *open* (fuel valve), which is now broadcast to the "Engine" process.

Once fuel is injected into the engine, the engine speed and temperature begin to increase. The engine can either stabilize at the throttle setting (which is the normal case), or the engine can accelerate and reach undesired speeds and temperatures (which is the abnormal case). From the point of view of the pilot, the transition out of motoring into either normal or high temperature states is non-deterministic. That is, the pilot cannot foresee when either state will happen (yet the pilot does know–based on historical data, training, and the mere existence of the abnormal procedure–that a faulty start may indeed occur). The uncertainty associated with reaching an undesired state, yet knowing that it may happen someday, is the foundation of Standard Operating Procedures and is exactly why procedures are in place.

In the event of a high engine temperature (HIGH TEMP), the procedure is to close the fuel valve, and as the engine fan rotates down, to re-engage the ground-starter (by setting the starter switch to GND). The procedure tells the pilot to motor the engine for at least 30 seconds, or continue beyond 30 seconds if the engine temperature is above 180 degrees. Once the temperature is below 180 degrees, the pilot should disengage the ground-starter.

Tasks and Specifications

The main objective of the pilot, once a high temperature occurs, is to remove the fuel source, quickly cool the engine, and bring it back to idle. In fact, if the pilot succeeds in cooling the engine following a hot start, the manual states that "maintenance personnel may be able to 'clear the item' and dispatch the airplane, depending on maximum EGT reached and its duration."

Therefore, the pilot's task goal is to "drive" the system back to the idle state. The specification is to perform this task as safely and efficiently as possible, minimizing damage to the engine and aircraft. It must be recognized, however, that the system (the engine and its surrounding environment) can interfere with this process and lead to sub-optimal configurations. Likewise, a pilot can, inadvertently, drive the system into an unwanted or sub-optimal configuration. (For example, see FAA Airworthiness Directive 88-07-02 that was issued in response to three documented cases in which Boeing B-767 pilots mistakenly shut down an engine during climb, while intending only to switch off a related sub-system). The model, therefore, should be expanded to account for pilot-initiated events that result in sub-optimal configurations.

Figure 4 is an expanded description of the model (omitting the "Fuel Control Switch" and "Engine Start Selector" process for brevity). Several states and transitions were added to account for pilot-initiated events and are discussed below:

Following a rapid rise in engine temperature (HIGH TEMP), if the pilot fails to take any action [t >= T], or the pilot mistakenly selects ground-starter (GND), the engine can be severely damaged. We denote this sub-optimal state as FAIL_D. If the pilot closes the fuel to the engine, but fails to motor the engine, this results in another sub-optimal state (FAIL_C). When the pilot selects ground-starter, but there is no oil pressure, the engine can be again severely damaged (FAIL_A). If there is enough oil pressure, but the fan speed is greater than 20 percent, engaging the starter may damage the starter (FAIL_B). Finally, when the engine is being motored and the pilot, by slip or mistake, disengages the starter before the mandatory 30 seconds or before the engine temperature is below 180 degrees, he or she may have to re-engage the ground-starter. This may not directly drive the system to a sub-optimal state, but it certainly takes more time and increases the potential for damage.

Now we can superimpose the task goal on the model. Our task is to drive the system to IDLE state, but we are willing to accept FAIL_A, if due to secondary effects (e.g., ruptured oil line) beyond our immediate control, unfortunate things occur. These two "acceptable" end-states are circled with a broken line in Figure 4. In contrast, FAIL_D, FAIL_C, and FAIL_B are not part of our task goals, but may happen if the pilot, for whatever reason, performs the wrong action.

Our specification called for efficiency and minimization of engine damage in the process of driving the engine to the IDLE state. If the pilot disengages the starter, before the mandatory 30 seconds or before the engine temperature is below 180 degrees, the task (drive the system to idle) is indeed achieved -- but our specification violated. The pilot may have to re-engage the ground-starter which will certainly takes more time and, as mentioned earlier, increase the potential for damage.

Analysis of Action Sequences

Now that we have superimposed the task goals and specifications on a model of the human-machine system, we can evaluate the procedure. Specifically, given the current system, we want to identify the safest and most efficient sequence of actions to get to the IDLE state. One way to trace

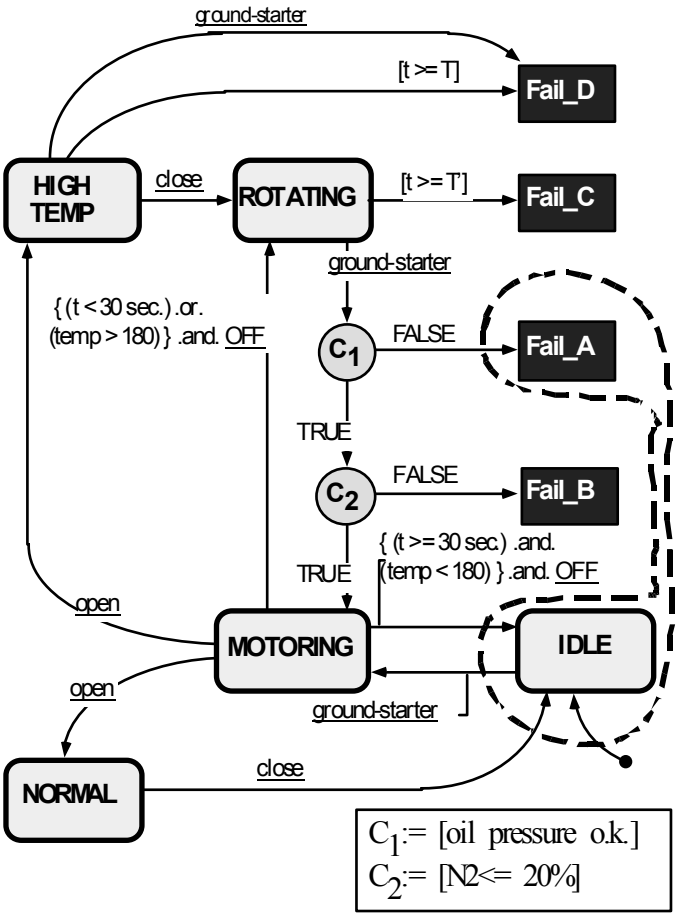

Figure 4. *Expanded model*.

this path is to "open up" the model in Figure 4 as a sequential tree of all possible actions. Figure 5 is a depiction of such a tree (focusing only on the relevant actions).

The nominal path from high temperature to idle traverses through rotating and motoring. Naturally, this sequence of actions must appear in the procedure. But we must also recognize that along the way there are numerous pitfalls to avoid, namely fail states A, B, C, and D. The procedure is the only aid to support the pilot along this "treacherous" path. Based on the model, the sequence of pilot actions following an irregular start (e.g., high temp state) is:

1. Close the fuel valve to the engine (the system transitions into rotating state).
2. Evaluate the situation: if oil pressure is O.K. and fan speed is at or below 20 percent, engage the ground-starter (and then transition to motoring), but if fan speed is above 20 percent, wait until it goes down; if there is no oil pressure, do not engage the starter (and transition to fail_A).
3. Motor for at least 30 seconds and monitor engine temperature. Continue motoring until the temperature goes below 180 degrees.

However, when we compare this sequence of actions and conditions with the procedure in Figure 2, we note an important discrepancy: The procedure, which must be executed immediately as a step-by-step sequence (with no time to think ahead), tells the pilot to close the fuel valve (fuel control switch–cutoff) and next immediately engage the

ground-starter (engine start selector–gnd). Only then (at the very end of the condition sentence and in small letters), does the procedure stipulate the oil pressure condition–"(unless no oil pressure)." Furthermore, only after three lines the procedure cautions the pilot not to engage the ground selector if N2 is above 20 percent. The order of statements in the procedure is not congruent with the order of actual events needed to drive the engine to idle. This sequential deficiency in this procedure may lead either to damaging the starter (FAIL_B) or to the even worse situation of damaging the engine because of no oil pressure while motoring (FAIL_A)!

There are also wording problems in this procedure: The multi-conditional sentence "Motor for 30 seconds or until EGT is below 180, whichever is longer (unless no oil pressure)" is difficult to comprehend. First, the two elements–"30 seconds" and "until EGT is below 180 degrees"–are difficult to equate; in the former time is explicit (30 seconds) and in the latter (until EGT is below 180 degrees) time is implicit (see Wickens, 1992 chap. 5; Bailey, 1989 pp. 363-367). Second, it is not clear whether "until EGT is below 180 degrees" includes or excludes the 30 seconds. Third, the logical operator .or. is followed by a selection criteria–"whichever is longer"-- which makes the sentence restrictive and therefore confusing. Furthermore, the ordering of the words in the sentence leads to reader to believe, initially, that he or she can either do A ("motor for 30 seconds") .or. B ("until EGT is below 180")--only then to be informed of the selection criteria ("whichever is longer") (see De Soto, London, and Handel, 1965 for a discussion on logical ordering). Fourth, the use of the term "longer" is confusing because the pilot is under the impression that he or she should try to "shorten" the time to motor the engine. The contextual conflict between these two opposing directions appears to mislead readers (see Laughery and Wogalter, 1997). Finally, from a grammatical point of view, there is a misuse of a parenthetical expression (unless no oil pressure). The use of parentheses implies that a logically remote relationship exists between the phrase within the parentheses and the rest of the sentence. In fact, the oil pressure is a pre-condition for the previous procedure step (engaging the ground-starter). The confusing wording of the conditional sentence may lead to situations in which the pilot may stop motoring prematurely (i.e., before the engine temperature drops below 180 degrees) or may simply not know what to do!

CONCLUSION

As mentioned earlier, it is a basic assumption in all high-risk industries that for known failures, the procedures supplied to the operators provide the "best" way to perform a given task. This is not the case with the procedure in Figure 2. We argue that the current process of designing and evaluating procedures can be much improved. New methods for describing such human-machine-environment interactions should replace the more intuitive and ad-hoc processes that are currently used by most high-risk industries. To assist in such a change, Human Factors researchers must develop design and evaluation methods to deal with far more complex systems and situations involving dynamics and timing constraints. Objective methods that will evaluate the vulnerability of a given procedure to human error are in great need. Can we predict where someone may fail in executing a procedure? Are there procedures that are more prone to human error? If so, why?

To conclude, while "deviation from operating procedures" is by far the highest-ranking crew-caused factor in aircraft accidents (NTSB, 1994), it is also true that many procedures are inherently incorrect. Such procedures "support" human error. Methods for more systematic, objective, and accurate procedure development in high-risk systems are desperately needed. We cannot continue to develop procedures for increasingly complex and automated systems while still using informal and ad-hoc techniques that are prone to error. This is a very important challenge in all high-risk industries.

REFERENCES

Bailey, R. W. (1989). *Human performance engineering: Using human factors/ergonomics to achieve computer system usability* (2nd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Degani, A., and Heymann, M. (1999). Pilot Autopilot Interaction: A formal perspective. In R. Jensen (Ed.), *Proceeding of the 10th Aviation Psychology Symposium*. Columbus, OH: Ohio State University.

Degani, A., and Wiener, E. L. (1993). Cockpit checklists: Concepts, design, and use. *Human Factors, 35*(2), 345-359.

Degani, A., and Wiener, E. L. (1997). Procedures in complex systems: The airline cockpit. *IEEE Transactions on Systems, Man, and Cybernetics, 27*(3), 302-312.

De Soto, C. B., London, M., and Handel, S. (1965). Social reasoning and spatial paralogic. *Journal of Personality and Social Psychology, 2*(4), 513-521.

Harel, D. (1988). On visual formalisms. *Communications of the ACM, 31*(5), 514-530.

Laughery, K. R., and Wogalter, M. S. (1997). Warning and risk perception. In G. Salvendy (Ed.), *Handbook of human factors and ergonomics* (pp. 1174-1198). New York: John Wiley.

Mosier, K. L., Palmer, E. A., and Degani, A. (1992). Electronic checklists: Implications for decision making. *Proceeding of the Human Factors Society 36th Annual Meeting* (pp. 7-11). Atlanta, GA: Human Factors Society.

National Transportation Safety Board (NTSB). (1994). *A review of flightcrew-involved major accidents of U.S. air carriers, 1978 through 1990* (Safety study, NTSB/SS-94/01). Washington, DC: Author.

R. J. Ramadge and W. M. Wonham, 1987, Supervisory control of a class of discrete event processes. *SIAM J. Control and Optimization, 25*(1), pp. 206-230.

Wickens, C. D. (1992). *Engineering psychology and human performance* (2nd ed.). New York: HarperCollins.

Wieringa, D., Moore, C., and Barnes, V. E. (1992). *Procedure Writing*. Piscataway, NJ: IEEE Press