Journal of Aviation/Aerospace Education & Research

Volume 33
Number 4 *Special Issue: Artificial Intelligence in Aviation*

Article 2

2024

Computational Cognitive Modeling of Pilot Performance in Pre-flight and Take-off Procedures

Rongbing Xu M.A.Sc
University of Waterloo, rongbing.xu@uwaterloo.ca

Shi Cao Ph.D.
University of Waterloo, shi.cao@uwaterloo.ca

Suzanne K. Kearns Ph.D.
University of Waterloo, suzanne.kearns@uwaterloo.ca

Ewa Niechwiej-Szwedo Ph.D.
University of Waterloo, eniechwiej@uwaterloo.ca

Elizabeth Irving Ph.D.
University of Waterloo, elizabeth.irving@uwaterloo.ca

Scholarly Commons Citation

Xu, R., Cao, S., Kearns, S. K., Niechwiej-Szwedo, E., & Irving, E. (2024). Computational Cognitive Modeling of Pilot Performance in Pre-flight and Take-off Procedures. *Journal of Aviation/Aerospace Education & Research, 33*(4). DOI: https://doi.org/10.58940/2329-258X.2026

JAAER

Computational Cognitive Modeling of Pilot Performance in Pre-Flight and Take-Off Procedures

Rongbing Xu[1a], **Shi Cao**[1b], **Suzanne K. Kearns**[1c], **Ewa Niechwiej-Szwedo**[1d], **Elizabeth Irving**[1e]
[1]University of Waterloo, ON N2L 3G1 Canada,
[a]rongbing.xu@uwaterloo.ca, [b]shi.cao@uwaterloo.ca, [c]suzanne.kearns@uwaterloo.ca, [d]eniechwiej@uwaterloo.ca,
[e]elizabeth.irving@uwaterloo.ca

Abstract

While the current practice of pilot training relies on flight instructors' subjective assessment, computational cognitive modeling may be used to support future objective assessment and diagnosis of pilot performance. We built two models in a cognitive architecture to simulate pilot flight performance during pre-flight and take-off tasks. Modeling results were compared with human results collected from the same tasks using X-Plane 11 flight simulator. The models were able to capture human pilot performance and workload results from both tasks with good levels of fitness (percentage errors ranging from 0.8% to 13.2%). This work demonstrated the capability and advantage of this theory-driven modeling approach for supporting general aviation pilot training. We expect that this type of cognitive model will be complementary to data-driven machine learning models, and the current work provides the foundation for future work to expand the modeling capability and test practical applications in general aviation.

Keywords: *Cognitive Modeling, QN-ACTR, Pilot Performance, Workload, Competency-Based Training*

Introduction

Current pilot training practice relies on flight instructors to assess trainees, diagnose their skill deficits, and provide recommendations for improvement. There have been discussions about the need for more objective methods of pilot assessment because flight instructors may miss important information and give inconsistent ratings, as they are limited to what can be observed visually rather than the comparably 'invisible' cognitive factors such as workload management and situation awareness (Brannick et al., 2002; Flin et al., 2018; Gontar & Hoermann, 2015; Sarter et al., 2017). It is therefore worthwhile to explore data-driven and model-driven computational approaches to complement human instructor assessment.

Machine learning has become a popular approach for pattern recognition and classification. Related previous studies in aviation are reviewed in the Background section. Previous studies have demonstrated that machine learning models have the potential to support pilot training by providing objective assessment of pilot performance; however, a large pool of pilot performance data is needed for model training and validation, which is not yet available in this research field. Another limitation of machine learning models is the skill diagnosis aspect of pilot training. While machine learning could provide performance classification, it is not suitable for providing skill analysis, diagnosis, or recommendation because machine learning models use a black-box approach, which only calculates the relationship between input and output, without explaining the mechanisms that generate pilot behaviors. For pilot training applications, there is a research gap and need for computational models that could provide insights to the cognitive activities of pilots. Addressing this limitation, our goal in the current study is to develop and test cognitive architecture models that reveal knowledge, skills, and cognitive mechanisms of pilot behaviors. We expect that such cognitive models can simulate pilot performance and support skill diagnosis and instruction to inform customized pilot training.

The current study uses the cognitive architecture modeling approach, which is both a unified theory of cognition and a computer simulation program representing human information processing (Taatgen & Anderson, 2010). Based on the results from simulation with different sets of competencies, researchers can establish the link between pilot performance deficits and the lack of specific knowledge and skills, and therefore training recommendations can be prescribed to focus on improving these aspects. More details of cognitive architectures are provided in the Background section.

Computational cognitive models could support competency-based pilot training. Competency-based training is a modern training approach that has been applied in many professional, vocational, and continuing education contexts (Hodge et al., 2020). It emphasizes the identification of competencies as explicit statements of knowledge, skills, and attitudes required to fully participate in a complex social practice and instructional practices that systematically built, assess, and retain the competencies (Kearns et al., 2017). Currently, the competence descriptions used in pilot training tend to be very general, and flight hours are the required metric used to assess expertise towards licensing criteria, rather than ratings on each competence statement. Instructor subjective evaluation is currently the major method to assess competence. Computational cognitive modeling can be a valuable tool to assist instructors and enhance students' training experience, by providing both a detailed symbolic representation of pilot knowledge and skills and a simulation platform for diagnosing causes of errors and performance deficits. Such cognitive models can also support the simulation of what-if scenarios, for purposes such as workload analysis and interface design evaluation, reducing the cost associated with conventional field testing and human-in-the-loop experiments.

This paper is organized as follows. The Background section provides a review of related existing work about both the machine learning approach and the cognitive architecture modeling approach in aviation. The Human Data Collection section describes our empirical study conducted to collect human data from pilots. Modeling and Simulation sections introduce model development, parameters, and assumptions. The Model Validation Section compares model results and human results. Overall findings and future considerations are presented in the Discussion and Conclusion section.

Background

Using flight data recording, simulators, and modeling techniques, researchers have developed and tested computational models for pilot performance assessment, such as rule-based exceedance models (Masiulionis et al., 2017), clustering models for abnormal detection (Lee et al., 2020; Li et al., 2015; Memarzadeh et al., 2020; Zhao et al., 2021), and supervised learning models for performance rating prediction (Palladino et al., 2021). For example, Fernández et al. (2019) applied clustering algorithms to 35,000 samples of flight approach operations in LEBL 25R. The dataset had 825 features including operation dynamics, adverse weather, aircraft configuration, pilot status, and flight static information. Applying Density-Based Spatial Clustering of Applications with Noise (DBSCAN), the researchers were able to identify outliers that exceeded the 95% quantile of the distribution. Similarly, a recent study showed that clustering algorithms can identify anomalies, such as minor bounced landing, which have not reached the parameter thresholds in exceedance detection definition specified by aviation authorities (e Silva & Murça, 2023). While most machine learning studies focused on landing anomalies and risk management, few studies have examined computational models to support pilot training. In a recent paper, Palladino et al., (2021) reported results from a proof-of-concept demonstration using data collected from a flight simulator. Data from 84 trials of four pilots included 61 operation dynamics and aircraft configuration variables with performance labels in three categories (good, fair, and poor). The authors compared different models, such as regression, stochastic gradient descent, and support vector machine classifiers, and the results showed that the support vector machine model had the best accuracy of 80%.

Researchers have been studying and developing cognitive architectures as an artificial intelligent method since 1970s (Anderson, 1974; Kotseruba et al., 2016; Taatgen & Anderson, 2010). Generally, a cognitive architecture has a framework representing different functional modules of the human information processing system. The interconnected modules, such as visual perception, long-term memory, production module, and manual control, send information and requests to each other and act collaboratively to solve a task problem. Each module has its processing logic, processing speed limitation, and capacity limitation, which are programmed in computer codes and parameters. While the architecture or framework represents general human capabilities and limitations, each model defined within the architecture simulates human performance in a specific task. Therefore, each model needs to describe task-specific knowledge and skills using symbolic representations such as chunks and production rules. A chunk represents a piece of declarative knowledge as a collection of related attributes and concepts. A production rule represents a piece of procedural skill as a collection of conditions and corresponding actions taken when the conditions are met. Differences in knowledge, skills, and parameters can reflect individual differences in strategy and competency. To build a cognitive architecture model, a researcher usually starts with task analysis that divides the task into multiple steps and elements. Each element is usually in the time scale of 50-200 ms to capture the detail of cognitive processes (Anderson et al., 2004). User studies are often needed to analyze an operator's flow of thoughts and decisions during task performance, and assumptions are needed to specify all the remaining details for the model. Once a model is built, it can be connected to a task simulation program via data protocols to run computer simulation and generate model behaviors that can be collected from the task simulator. The model's internal processing can also be recorded for analysis, for example, for the prediction of workload. After each task component model is validated, multiple component models can be combined to represent the entire task. The modeling work using cognitive architectures is additive, which means later work can be built upon previously tested model components and parameters. A cognitive architecture is implemented as a computer simulation program, so it can be connected to a flight simulator to perform model-in-the-loop simulation and generate model predictions. To validate a model, researchers usually need to collect human data from various task conditions using the same task simulator as used by the model. Then, model results are compared with human results to analyze model fitness. Good fitness is reflected by small errors (discrepancies) and a similar trend or pattern observable from different task conditions in both model and human results. The goal is to let the model faithfully represent the capability and limitation reflected in the human data, which could be from novice or experienced pilots. Previous studies have built cognitive architecture models for some aspects of pilot performance in a limited number of tasks. The following paragraphs give an overview of the major cognitive architectures used in modeling aviation tasks.

Adaptive Control of Thought-Rational (ACT-R) is a widely used cognitive architecture that has influenced many other cognitive architectures. ACT-R's modules include procedural, intentional/goal, declarative, imaginal, visual, aural, motor, and speech modules. It has unique computational mechanisms for modeling memory and learning as changes in the activation level of chunks and the utility value of production rules (Anderson et al., 2004). Many experiments and modeling work have accumulated a collection of tested parameters and validated models for tasks such as memory retrieval, visual attention, and perceptual-motor responses. Recent work on Queueing Network-Adaptive Control of Thought Rational (QN-ACTR) has added queueing mechanisms to ACT-R to support the scheduling and natural processing of multiple streams of task information within the architecture (Cao & Liu, 2013b). The validated modeling mechanisms and existing models can support future work using similar task components and assumptions. Using ACT-R, previous work has built models for manual control in aircraft taxiing tasks (Schoelles & Gray, 2011) and visual attention distribution during approach and landing with different types of information display systems (Byrne et al., 2008).

Cognitive Architecture for Safety Critical Task Simulation (CASCaS) is a recent cognitive architecture that combines production rule and control theory systems (Lüdtke et al., 2009). The production rule system in CASCaS is like ACT-R. CASCaS does not have sophisticated declarative memory and learning mechanisms, but it has an autonomous behavior mechanism that is assumed to work in parallel with the production rule system (Lüdtke et al., 2011). The autonomous behavior mechanism implements control theory equations to model dynamic vehicle control behavior. CASCaS focuses on modeling human errors resulting from both learned carelessness and attention allocation issues. Researchers have demonstrated CASCaS in modeling pilot errors and error recovery behavior in flight management tasks due to issues with the perception of automatic flight mode changes (Lüdtke et al., 2010).

Soar is a cognitive architecture that has similar modules as ACT-R (Laird et al., 1987). Previously Soar had only production rule knowledge without declarative knowledge, but recent work has added declarative knowledge representation and computations as inspired by ACT-R. Soar modeling focuses on producing intelligent cognitive processes such as situation analysis, decision making, and problem solving. The goal of Soar modeling is usually for the design of highly intelligent agents rather than representing human limitations. Using Soar, researchers have modelled intelligent pilots for both fixed-wing and rotary-wing aircraft simulations (Hill et al., 1998; Jones et al., 1999).

Air Man-Machine Integrated Design and Analysis System (Air MIDAS) is an integrated tool for representing and analyzing human-machine performance for aircraft interface design (Verma et al., 2003). NASA has utilized Air MIDAS extensively in the study of a variety of human-machine systems (Ferrin et al., 1988). Compared with ACT-R, Air MIDAS does not aim to represent all cognitive mechanisms such as long-term memory, learning, and decision making. Instead, Air MIDAS uses cognitive engineering methods, such as task analysis, task scheduling, expert evaluation, and probabilistic models to simulate human performance and workload (Gore et al., 2008). It also has integrated physical models for the environment, aircraft cockpit, and pilot anthropometry to support interface design needs. An Air MIDAS model is composed of three components: input, output, and processing. Input includes the external environment and data as well as the current task; processing includes all input data and rule matching; output includes the current state of the model and its performance data (Sebok et al., 2013). Previous studies have compared Air MIDAS and ACT-R in the modeling of skilled pilots performing commercial aircraft approach and landing using a new design of synthetic vision systems (Leiden & Best, 2005). The results showed that Air MIDAS models of visual attention fixations had r-squared fitness of about 55% on average, whereas ACT-R had r-squared fitness of around 84% on average.

In summary, previous studies have built and examined a limited number of cognitive architecture models for the simulation of pilot performance with many limitations. Previous studies focused on limited human factors measures, such as visual attention distribution and taxiing control performance (Byrne et al., 2004; Schoelles & Gray, 2011), and there is a lack of simulation of flight control performance (e.g., altitude and pitch angle), workload, and situation awareness. The existing publications did not provide any detail of task-specific knowledge and strategies, such as production rules (Byrne et al., 2008). They did not report any detailed validation data comparing model results and human results (Schoelles & Gray, 2011). Researchers have acknowledged the challenges in modeling various aspects of pilot performance because many production rules must be programmed to represent human complex strategies and decision-making processes (Byrne et al., 2008). To address these limitations, we will study computational cognitive modeling of pilot performance from a unique angle focusing on initial skill training stages in general aviation, where the degree of skill requirements is simpler than commercial airline flights. In the current study, we start our work with pre-flight and take-off procedures of small aircraft that generally have fewer steps with lower complexity. We used the ACT-R approach because previous work has accumulated the most production rules and modeling examples in ACT-R related to the current tasks. Detailed human data during flight operations are collected and compared with model data from interacting with the same flight simulator to validate the models. We publish our data and modeling details to support future work along the same direction of research for the development of competency-based pilot training technologies.

Human Data Collection

Using a flight simulator, human operation data were collected, including manual control actions and their timing, aircraft attitude and dynamics, and pilot workload.

Participants

We recruited 18 student pilots between 18 and 24 years old from the University of Waterloo. Although we tried to recruit both males and females, all the participants were male due to the low number of female pilots in aviation. They had on average 68 hours of flight experience (SD = 7.3). All had flight experience at the Region of Waterloo International Airport (CYKF) with the Cessna 172SP aircraft that was used in the flight simulation. All had passed the Transport Canada category 1 civil aviation medical exam.

Apparatus

The simulator system consisted of various displays and controllers (Figure 1), including a yoke and throttle quadrant, a rudder paddle, a flight multi-panel, a radio panel, a flight switch panel, and six flight instrument panels (all made by Logitech). Three 24-inch monitors were used. X-Plane 11 was used as the flight simulation software. We used X-Plane's dataref function to monitor the status of the aircraft, with data recording rate at 20 Hz. The aircraft model was a Cessna 172SP (model developed by Airfoillabs). A checklist (see Appendix A) describing pre-flight and takeoff procedures was provided to the participants. Each checklist item was extracted from the actual aircraft's pilot checklist.

Task and Scenario

The simulated environment started at the Region of Waterloo International Airport (CYKF), which has an elevation of 1,055 feet Above Sea Level (ASL). The weather was calm and sunny with a temperature of 24°C at 9 AM Eastern Daylight Time. The runways were completely dry, with excellent visibility. The air pressure was 29.92 inches of mercury. There are seven aprons at the airport. The test aircraft was parked on Apron 3. In Task 1, participants were instructed to prepare the aircraft using the pre-flight checklist provided. Participants were required to monitor the aircraft's condition and determine when to take the next action based on their knowledge. Task 2 required participants to take off from the airport. The instruction was to take off on runway 08, climb to 4,000 feet (ASL), and maintain a heading of 74 degrees while climbing. Participants were instructed to follow the take-off checklist, maintain full throttle and rich mixture during takeoff, and not use any trim.

Procedure

Before the study, participants completed a consent form and a demographic questionnaire. This study was reviewed and approved by a research ethics committee at the University of Waterloo. All participants were given about ten minutes to practice in a scenario that was different from the actual scenario used in the formal task. It served to familiarize participants with the flight controllers and the simulator. Participants were provided with sample instructions to ensure that they could follow them and complete the task safely and accurately, just as they would if they were piloting a real aircraft. After the practice session, the participants were asked to sequentially complete the two tasks: pre-flight preparation and takeoff procedures. There was one trial for each task. All participants were given a 5-minute break between the two tasks. Before the start of each trial, all switches on the instrument panel were consistently pre-set at the same setting for all participants.

Throughout the study, we recorded the simulator's flight dynamics data and switch/button states. Participants' actions were video recorded for the analysis of response time and intervals between actions. A camera was used to record the participant's hand movement on the flight control panel, and another camera was used to record the entire panel and displays. Participants completed a NASA-TLX questionnaire after each task to measure subjective workload.

Human Data

For the checklist procedure in Task 1, the time duration used for each step was analyzed from the video recordings. The time duration may include three types: reading, checking, and acting. Reading refers to the time required for participants to read and comprehend the checklist items printed on the checklist. Checking refers to the time required for participants to verify the status of aircraft components by visually examining the control panel and devices. Acting refers to the time required for participants to move their hands and manipulate aircraft components. The average time durations for each checklist item from the human participants were calculated and shown in Table 1. The total time for human participants to complete all the checklist items was 64.13 s on average (SD = 5.18 s).

During Task 2, participants needed to continuously control the aircraft's airspeed and attitude. The total time to complete the takeoff and climb procedure was recorded. As shown in Figure **??**, while most participants completed it within 220-260 s, five participants used a longer time of around 280-320 s, which was the result of deviation from the standard operation procedures possibly due to the lack of skills or the application of non-standard strategies.

We compared the flight experience hours between the two groups. T test showed a significant difference, $t(16) = 2.209$, $p = 0.042$. The five participants with longer task time had fewer flight hours (44.4 hours, SD = 13.6) than the rest of the participants (77.1 hours, SD = 31.5). In the current study, our focus is to demonstrate the modeling approach in an example of standard task performance. As a result, we treated the five participants with total time longer than 270 s as outliers, and the model was designed to only represent the standard operation and strategy in the current study. Since in cognitive architecture models, different skills and strategies could be modelled as different production rules, we will further explore the modeling of the outlier group in future studies. For the remaining 13 participants, the average of their takeoff and climb total time was 231.9 s (SD = 8.2 s). Their major flight performance results are summarized in Table 2.

Regarding overall subjective workload measured by NASA-TLX (possible range from 0 to 100), the average human results were 25.0 (SD = 7.4) for Task 1 (pre-flight) and 40.4 (SD = 14.5) for Task 2 (take-off). The workload from Task 1 is significantly lower than Task 2, $t(12) = -4.94$, p ¡ 0.001. These human data provided the references for model tuning and validation.

Modelling & Simulation

The QN-ACTR cognitive architecture was used in the current study because previous work has demonstrated its capabilities and advantages in simulating and predicting multitasking behaviors, workload, and situation awareness (Cao & Liu, 2011, 2012, 2013a; Rehman et al., 2019; Xu & Cao, 2021), which allow us to comprehensively model different human factor aspects of pilot flight behavior in the current study as well as planed future work. To build the models for both Task 1 and Task 2, we first analyzed the steps and task elements that pilots need to perform for each task following the checklists. Then procedures from these steps were programmed as production rules (condition-action pairs) that represent skills used by the model to complete the tasks. The QN-ACTR models running these production rules can be directly connected to the same flight simulator (X-Plane 11) as used by the human participants. Through the model-in-the-loop simulation, performance variables and workload predictions were generated. We compared model and human results to examine model fitness.

QN-ACTR and Connection to X-Plane

The module structure of QN-ACTR is shown in Figure 3. Recent studies in QN-ACTR have improved its abilities to model and simulate workload (Cao & Liu, 2013b), situation awareness (Rehman et al., 2019), and human performance in takeover from automation control (Deng et al., 2019). These capabilities can support future work to comprehensively simulate pilot performance and other human factor aspects. Source code and sample models for QN-ACTR can be found online (https://github.com/HOMlab).

The flight simulation for the current study was conducted with X-Plane 11 and a Cessna 172SP aircraft model plug-in. As a simulation software certified by the FAA for pilot training, X-Plane is widely used in scientific research and commercial development. A pilot model in QN-ACTR can receive and send data to X-Plane in real time via the User Datagram Protocol (UDP) interface. This data link enables the model to assess the aircraft's and external status environment and send control directives to the simulated aircraft. The model-in-the-loop simulation is executed as a control loop via a UDP connection between the cognitive model in QN-ACTR (Java programming language) and X-Plane 11, running on the same computer. X-Plane will continuously update the model with aircraft and environmental information. The model will adhere to its own production rules, perform information processing, and issue control commands. Commands, such as pressing a button or turning the yoke, are converted to data compatible with X-Plane and returned to the simulator. Using this control loop, the piloting simulation is done as a discrete event simulation of the pilot and the aircraft in the virtual environment.

To support the connection between the cognitive model and X-Plane code, the X-Plane Connect (XPC) software package was utilized (https://github.com/nasa/XplaneConnect). Developed by NASA, XPC is an open-source research tool that enables X-Plane code to communicate with external code using C, C++, Java, MATLAB, or Python functions. We tested the connectivity between QN-ACTR models and X-Plane using XPC and confirmed that the delay introduced by XPC connection was neglectable and did not affect the performance of the model.

Model Design

The modeling work involved the following steps, which are a common and standard approach used in cognitive architecture modeling work. First, a cognitive task analysis was conducted. Flight checklists and typical flight instructions for the pre-flight and take-off tasks were consulted for the cognitive task analysis. Second, the sequence of steps was coded into a series of production rules. Third, model parameters were determined for each model component or sub-task procedure. Most parameter values used their default values that have been used and validated in previous work. Some parameter values such as hand motion time when pressing buttons on the cockpit control panel and coefficients in the manual control equations for flying the aircraft were estimated and adjusted by the researchers to fit the current human data because previous work has not validated these parameters for the same tasks. After these steps, the finalized models were examined in the simulation of Task 1 and Task 2, and the overall performance and workload results were compared with corresponding human data from the same scenarios.

We designed the overall flight control strategy of the model to include three major components including monitoring, decision-making, and control (Figure 4). This general design follows a similar high-level strategy used in previous modeling work of dynamic vehicle control (Deng et al., 2019; Salvucci, 2006). The monitoring component involves attending to the environment and displays, perceiving external information via the visual module, and preparing the information for decision making. The decision-making component involves the procedural module making decisions and selecting responses prior to generating actions via production-rule pattern matching. The control component involves the motor module receiving action commands and carrying out motor control operations. Human reaction time is reflected in model parameters that represent the time duration needed for each module to complete its processing. For example, the execution of each production rule in the procedural module takes 50 ms by default (Anderson et al., 2004).

Specifically, the model is built to control the aircraft through several types of control actions.

- **Switches and Buttons:** Interaction between the model and the buttons, or switches, is a form of discrete control actions. When a target button or switch needs to be pressed or changed, the model performs the following three general steps: shifting visual attention to the target, moving the hand to the target, and then changing the status of the button or switch.
- **Throttle and Mixture:** When the model needs to adjust the throttle or mixture, it issues a continuous action to gradually change the throttle or mixture position from its current value to the new desired value.
- **Yoke and Rudder:** During take-off under Visual Flight Rules, the model keeps observing the environment to determine the aircraft's position and attitude. The model continuously adjusts the yoke and rudder paddle to maintain the desired attitude, based on the perceived pitch, roll, and heading angles.
- **Steering on the Ground:** When steering an aircraft during taxiing, the model keeps the visual attention on the taxi line in front of the aircraft to determine the aircraft's position and adjust the pedal accordingly as a continuous action. Since the tasks simulated in the current study did not include taxiing, steering related production rules and model mechanisms were not used in the current study.

At the implementation level, the above control actions were programmed as two types of operations. First, the x-plane-discrete-action operation applies to all interactions with buttons and switches. Typically, this is a one-time operation, such as turning the beacon light on, because it is not needed to repeat these actions once done properly. When the model executes this operation, it sends a single command to X-Plane and sets the corresponding variable to its new value. Second, the x-plane-continuous-action operation applies to all interactions with the yoke, pedal, throttle, and mixture. Each continuous action is implemented as a series of micro steps. For throttle and mixture control actions, the micro steps are implemented at 20 Hz (50 ms for each step). For yoke, rudder, and steering control actions, each micro step of control typically takes 150 ms (executing three production rules).

Pre-Flight Model (Task 1)

The model for the pre-flight task was relatively simple because the pre-flight checklist provides an explicit and clear sequence of steps. For each checklist item, the model would visually scan the checklist to read the information, visually scan the gauges or controllers to confirm its status, and then perform an action to make any adjustment if needed. The production rules and their descriptions are provided in Appendix B. For the model to simulate interaction with the control panel, the model's visual module and motor module need to know the relative locations of different gauges and controllers on the control panel. Since this information was not available from the X-Plane data protocol, we created a virtual display representing the control panel layout by utilizing existing modeling mechanisms for computer display interaction built in QN-ACTR. The virtual layout and the relative sizes of objects represented the same physical system used in the simulator setup for human participants (Figure 5). The virtual display representing the aircraft control panel area was set as 2,000 by 1,100 pixels, and 1 cm in the physical setup corresponds to 20 pixels in the virtual display.

In the pre-flight task model, only three parameters were adjusted (Table 3), and all other parameters used their default values in the cognitive architecture. The imaginal-delay parameter represents the time duration needed for the model to form the understanding and meaning of visual objects it sees; for example, when reading a checklist item or examining switch states, we used an estimated 1.0 s for all kinds of representations. The hand motion time for reaching and pressing a button or switch was estimated as 1.0 s, and the time for the hand to move back to its original position was also estimated as 1.0 s. These estimated values were used to represent average human performance.

Take-Off Model (Task 2)

In the take-off task, pilots' operation mainly involves controlling the aircraft using the yoke and rudder pedal. This operation was simulated as a control loop including both the pilot model and the aircraft in the fight simulator. Within a typical control cycle, the model used three production rules to first monitor aircraft attitude (pitch, roll, and yaw), then encode their values, and finally issue control actions of minor adjustments using the controllers (yoke and rudder pedal). The production rules were designed following the take-off checklist items and previous modeling conventions (see Appendix B). The model followed the checklist to apply full throttle while take-off.

Regarding the manual control of aircraft attitude, mathematical equations were used to determine the amount of controller adjustment given the model's perceived value of flight parameter discrepancy, which was the difference between the desired value (e.g., target pitch angle for take-off) and their current value (e.g., current pitch angle). The equations were defined by following a method of discrete control laws used in a previous model of vehicle control (Salvucci & Gray, 2004). Three equations were used for the three dimensions of pitch, roll, and yaw, respectively.

$$\Delta\varphi_{\text{yoke-pull}} = k_y \Delta\theta_{\text{pitch}} + k_{yl}\theta_{\text{pitch}}\Delta t, \quad (1)$$

$$\Delta\varphi_{\text{yoke-steer}} = k_s \Delta\theta_{\text{roll}} + k_{sl}\theta_{\text{roll}}\Delta t, \quad (2)$$

$$\Delta\varphi_{\text{rudder}} = k_r \Delta\theta_{\text{yaw}} + k_{rl}\theta_{\text{yaw}}\Delta t. \quad (3)$$

In these equations, Δt is the control cycle (150 ms, executing three production rules); θ is the flight parameter discrepancy (for pitch, roll, and yaw respectively); $\Delta\theta$ is the change of θ within Δt. $\Delta\varphi_{\text{yoke-pull}}$, $\Delta\varphi_{\text{yoke-steer}}$, and $\Delta\varphi_{\text{rudder}}$ are the micro adjustments to the controllers applied within each control cycle; k values are coefficient constants for each term in the equations. The model was designed to maintain a stable take-off (target roll is 0 degrees), following Task 2 instructions (maintain heading 74 degrees) and common general aviation training practice. Specifically, during initial takeoff, the model applies no yoke input until the airspeed reaches 55 knots. Once it reaches 55 knots, the model starts to apply yoke adjustment (as described by Equations (1)-(3)) to gradually reach a target pitch angle of 7.5 degrees. Once the aircraft has left the ground (above ground level altitude over 150 feet), the model keeps monitoring the airspeed while climbing. If the airspeed is faster than desired (faster than 75 knots), the model applies yoke adjustment to gradually reach a larger pitch angle (estimated as 10 degrees), so the airspeed will decrease. If the airspeed is slower than desired (slower than 75 knots), the model applies yoke adjustment to gradually reach a smaller pitch angle (estimated as 5 degrees), so the airspeed will increase. We understand this control logic used by the model is rigid and specific, which does not represent variations within each participant or between participants. However, it provides a simple modeling solution to demonstrate the modeling capability of our approach. Future studies will further improve the modeling detail to increase model fitness and represent performance variations, for example, by drawing parameter values from distributions. To determine the coefficient k values, we conducted multiple tests with different values. The tests showed that when k is held constant, larger values of k_I lead to faster recovery time of the aircraft's attitude. When k_I is held constant, larger values of k lead to greater fluctuation of the aircraft's attitude. In the end, we found that k values = 35 and k_I values = 15 for all three questions could produce good model fitness, so these values were used.

Model Validation

For model validation, we connected the models to the X-Plane simulator under the same configurations as we did for the human tests. Model simulations were repeated ten times to obtain average model results to be compared with average human results. Since the models were not designed to simulate human variability or individual difference, the model results did not contain a large variance, and ten runs were enough to generate reliable estimation of average model results. Model fitness was evaluated using Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE), where X_h represents the human data, X_m means the model data, i is the index, and n is the number of comparisons made. Smaller RMSE and MAPE mean better model fitness.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{hi} - X_{mi})^2}{n}} \quad \text{and} \tag{4}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{X_{hi} - X_{mi}}{X_{hi}}\right|, \tag{5}$$

For the pre-flight task, we recorded the model's average time to complete each checklist item, as shown in Table 4.

For Task 1, the total duration of the model's run was 69.84 seconds on average. The total time from the human data was 64.13 s on average, so the RMSE is 5.71 s, and the MAPE is 8.9%. For Task 2 flight performance measures, the comparison and model fitness results are shown in Table 5.

In addition to simulating pilot performance, QN-ACTR can also assess pilot workload by analyzing total server utilization, which represents how busy the architecture's modules are processing information during each task (Cao & Liu, 2011). The overall utilization value ranges from 0.0 (completely idle) to 1.0 (completely loaded). While server utilization reflects the theoretical absolute value of workload, NASA-TLX is a subjective report measure that better reflects relative workload between conditions rather than the theoretical absolute value. As a result, for model fitness analysis, we test whether the ratio of workload between Task 2 and Task 1 predicted by the model is like the ratio from the NASA-TLX results reported by the participants, which showed Task 2 had higher workload than Task 1. From the simulation results, the model had overall utilization of 0.22 in the take-off task and 0.13 in the pre-flight checklist task, so the ratio is 1.69. From the human NASA-TLX results, this ratio is 1.62. Therefore, the model fitness has RMSE = 0.07, and MAPE = 4.3%.

Discussion and Conclusion

In the current study, we applied computational cognitive models built using QN-ACTR to simulate pilot behavior during pre-flight and take-off tasks. The models' task specific knowledge and skills were coded by analyzing task procedures and aircraft checklists. With a total of seven parameters adjusted, and the other parameters using default values in the cognitive architecture, the models were able to capture human pilot performance and workload results from both the pre-flight and take-off tasks with good levels of fit (MAPE ranging from 0.8% to 13.2%).

In contrast to machine learning models that apply a black-box approach, the current modeling approach is a transparent glass-box approach because the model defines and simulates the cognitive mechanisms, knowledge, and skills that collectively produce human pilot behavior. The two approaches are complementary for research and industrial applications; however, existing work studying pilot behavior mostly followed the black-box approach. Although the glass-box approach has been used to study driver behavior (Deng et al., 2019; Salvucci & Gray, 2004), the current study is the first one to publish model building and validation details for general aviation pre-flight and take-off tasks using a cognitive architecture.

A major benefit of the computational cognitive model is that the model's behavior, capability, and lack of performance are explainable and understandable to human beings. These features are very valuable to pilot training research and practice because instructors need better ways to model and diagnose trainees' skill development. We believe that further advancing computational cognitive models will support competency-based pilot training. Cognitive modeling provides a method to objectively ascertain the knowledge and skills that define pilot competencies, which can go deeper beyond the current descriptive competence statements obtained from expert subjective opinions. By building cognitive models of skilled expert pilots, researchers can obtain the optimal knowledge and skills for setting training targets. Then by modeling trainees and comparing trainee models with expert models, researchers can identify trainees' knowledge and skill deficiencies in each specific task and develop personalized training scenarios to target individual improvement towards licensing standards.

Machine learning models and cognitive architecture models are both computational models of human performance, and they could be used together to support the next generation of pilot training. World demand for commercial pilots is estimated to grow significantly. Given that demand is expected to exceed pilot training capacity, the International Civil Aviation Organization (ICAO) has designated retaining the next generation of aviation professionals as a "global priority." The limited number of certified flight instructors is a bottleneck in training capacity. As a result, objective flight training methods using data and computational models are needed. Future studies will need to collect pilot training data with flight data recordings and instructor assessment. The data will support the training of data-driven machine learning models and the validation of theory-driven cognitive models. Combining the advantages of both modeling approaches, future models could support both objective assessment and diagnosis of pilot performance.

Envisioning the future when computational models are validated, we expect that computerized tools can be developed to facilitate training. A performance assessment tool can take a student's flight data and control action data from a simulator or aircraft as the input and generate performance assessment as the output. For example, landing performance can be graded by the tool on a 4-point scale following a similar rubric as currently used by instructors. A diagnostic tool can tell students why performance is poor in certain aspects and which piece of skill or knowledge is lacking. For example, the tool may say that landing performance is poor because the round out action is too late and the rate of yoke pulling is too fast. The model knows this because many simulations are performed to create a library of poor performance caused by different types of skill deficits. A feedback tool can give students suggestions about how to improve. The suggestions will be based on human instructor recommendations and verified via model simulation.

While the current work focused on performance measures, such as altitude, airspeed, heading, control column input, and reaction time, it is possible for a cognitive architecture model to simulate mental workload using the concept of server utilization and predict situation awareness using what the model sees and does not see. These topics are beyond the scope of this paper. Currently, the cognitive modelling of workload and situation awareness is less developed than performance models. In contrast, wearable physiological sensors such as heart rate monitors and eye trackers may provide better data sources for the prediction of workload and situation awareness using machine learning models, which may support the design of useful tools in the future.

Since the current study is an initial proof-of-concept demonstration, there are many limitations that need to be addressed in future work before cognitive models can be applied to pilot training practice. First, we only tested two simple tasks in ideal conditions, future studies will need to expand the tasks and weather conditions. In addition to standard operating procedures, emergency operating procedures can also be tested and modeled using cognitive architecture models. Second, the current models did not consider individual difference factors such as visual acuity, perceptual speed, risk tolerance, and motor response accuracy and speed. These factors may be added into the model in future work by identifying and confirming the relationship between these factors and the corresponding model parameter values. For example, motor speed may be controlled by the motor execution duration parameter. Regarding the outlier group in the current human results (the five participants with longer task time), future studies could explore alternative production rules and modified model parameters as ways to model their difference. Third, there are other human factors such as stress and fatigue that can affect pilot performance. For these aspects, existing literature in cognitive modeling has accumulated some methods to explain and simulate their effects on performance (Khosroshahi et al., 2019). Future studies could integrate these methods into the pilot model and validate the model performance. Finally, there is an inherent variability of human behavior that was not captured by the current model. People's visual scan pattern, decision strategy, response speed, and manual control accuracy are not perfectly consistent but vary sometimes. The same person's performance changes as they repeat the same task multiple times. To capture this finer detail of human behavior, future work may need to add a random offset value to some model parameters or create some alternative production rules that will have some probabilities to be used for the same procedure.

In conclusion, we developed and validated two computational cognitive models using a cognitive architecture to simulate pilot flight performance and workload in pre-flight and take-off tasks flying a Cessna 172SP aircraft. This work demonstrated the capability and advantage of this theory-driven glass-box modeling approach for supporting general aviation pilot training. By collecting more human data and further developing models for other tasks and conditions, researchers can combine the efforts to create a tool for pilot skill diagnosis and support competency-based pilot training. We expect that these types of cognitive models will be complementary to data-driven machine learning models, and the current work provides the foundation for future work to expand the modeling capability and test practical applications in general aviation.

Author Note

This work was supported by an NSERC Discovery Grant (RGPIN-2015-04134), New Frontiers in Research Foundation Exploration Grant (NFRFE-2020-00753), and the Waterloo Institute for Sustainable Aeronautics (WISA) Research-for-Impact program funded by the Federal Economic Development Agency for Southern Ontario (FedDev). This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the University of Waterloo Office of Research Ethics Committee.

References

Anderson, J. R. (1974). Verbatim and propositional representation of sentences in immediate and long-term memory. *Journal of Verbal Learning and*

Verbal Behavior, *13*(2), 149–162. https://doi.org/10.1016/S0022-5371(74)80039-3

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036–1060. https://doi.org/10.1037/0033-295X.111.4.1036

Brannick, M. T., Prince, C., & Salas, E. (2002). The reliability of instructor evaluations of crew performance: Good news and not so good news. *The International Journal of Aviation Psychology*, *12*(3), 241–261. https://doi.org/10.1207/S15327108IJAP1203_4

Byrne, M. D., Kirlik, A., Allard, T., Hooey, B. L., Gluck, K. A., Wickens, C. D., Pritchett, A. R., & Foyle, D. C. (2008). Issues and challenges in human performance modeling in aviation: Goals, advances, and gaps. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *52*(13), 926–929. https://doi.org/10.1177/154193120805201306

Byrne, M. D., Kirlik, A., Fleetwood, M. D., Huss, D. G., Kosorukoff, A., Lin, R., & Fick, C. S. (2004). A closed-loop, act-r approach to modeling approach and landing with and without synthetic vision system (svs) technology. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *48*, 2111–2115. https://doi.org/10.1177/154193120404801707

Cao, S., & Liu, Y. (2011). Mental workload modeling in an integrated cognitive architecture. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *55*(1), 2083–2087. https://doi.org/10.1177/1071181311551434

Cao, S., & Liu, Y. (2012). Qn-actr modeling of multitask performance of dynamic and complex cognitive tasks. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *56*(1), 1015–1019. https://doi.org/10.1177/1071181312561212

Cao, S., & Liu, Y. (2013a). Queueing network-actr modeling of concurrent tasks involving multiple controlled processes. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *57*(1), 768–772. https://doi.org/10.1177/1541931213571168

Cao, S., & Liu, Y. (2013b). Queueing network-adaptive control of thought rational (qn-actr): An integrated cognitive architecture for modelling complex cognitive and multi-task performance. *International Journal of Human Factors Modelling and Simulation*, *4*(1), 63–86. https://doi.org/10.1504/IJHFMS.2013.055790

Deng, C., Cao, S., Wu, C., & Lyu, N. (2019). Modeling driver take-over reaction time and emergency response time using an integrated cognitive architecture. *Transportation Research Record: Journal of the Transportation Research Board*, *2673*(12), 380–390. https://doi.org/10.1177/0361198119842114

e Silva, L. C., & Murça, M. C. R. (2023). A data analytics framework for anomaly detection in flight operations. *Journal of Air Transport Management*, *110*, 102409. https://doi.org/10.1016/j.jairtraman.2023.102409

Fernández, A., Martınez, D., Hernández, P., Cristóbal, S., Schwaiger, F., Nunez, J. M., & Ruiz, J. M. (2019). Flight data monitoring (fdm) unknown hazards detection during approach phase using clustering techniques and autoencoders. *Proceedings of the Ninth SESAR Innovation Days*, 2–5.

Ferrin, T. E., Huang, C. C., Jarvis, L. E., & Langridge, R. (1988). The midas display system. *Journal of Molecular Graphics*, *6*(1), 13–27. https://doi.org/10.1016/0263-7855(88)80054-7

Flin, R., Martin, L., Goeters, K.-M., Hörmann, H.-J., Amalberti, R., Valot, C., & Nijhuis, H. (2018). Development of the notechs (non-technical skills) system for assessing pilots' crm skills. In H. Muir & D. Harris (Eds.), *Human factors and aerospace safety* (pp. 97–119). Routledge. https://doi.org/10.4324/9781315194035-1

Gontar, P., & Hoermann, H.-J. (2015). Interrater reliability at the top end: Measures of pilots' nontechnical performance. *The International Journal of Aviation Psychology*, *25*(3–4), 171–190. https://doi.org/10.1080/10508414.2015.1162636

Gore, B. F., Hooey, B. L., Foyle, D. C., & Scott-Nash, S. (2008). Meeting the challenge of cognitive human performance model interpretability though transparency: Midas v5.x. *2nd International Conference on Applied Human Factors and Ergonomics*. https://hsi.arc.nasa.gov/groups/midas/documents/Gore-Hooey-Foyle-SSN-ref.pdf

Hill, R., Chen, J., Gratch, J., Rosenbloom, P., & Tambe, M. (1998). Soar-rwa: Planning, teamwork, and intelligent behavior for synthetic rotary wing aircraft. *7th Conference on Computer Generated Forces & Behavioral Representation*. https://projects.iq.harvard.edu/sites/projects.iq.harvard.edu/files/teamcore/files/1998_11_teamcore_cgf_98.pdf

Hodge, S., Mavin, T., & Kearns, S. (2020). Hermeneutic dimensions of competency-based education and training. *Vocations and Learning*, *13*(1), 27–46. https://doi.org/10.1007/s12186-019-09227-y

Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P., & Koss, F. V. (1999). Automated intelligent pilots for combat flight simulation. *AI Magazine*, *20*(1), 27–27.

Kearns, S. K., Mavin, T. J., & Hodge, S. (2017). *Competency-based education in aviation: Ex-*

ploring alternate training pathways. Routledge. https://doi.org/10.4324/9781315563220

Khosroshahi, E., Salvucci, D. D., Gunzelmann, G., & Veksler, B. (2019). A unified model of fatigue in a cognitive architecture: Time-of-day and time-on-task effects on task performance. *CogSci*, 567–573.

Kotseruba, I., Gonzalez, O. J. A., & Tsotsos, J. K. (2016). A review of 40 years of cognitive architecture research: Focus on perception, attention, learning and applications. *ArXiv*, 1–74.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1–64. https://doi.org/10.1016/0004-3702(87)90050-6

Lee, H., Li, G., Rai, A., & Chattopadhyay, A. (2020). Real-time anomaly detection framework using a support vector regression for the safety monitoring of commercial aircraft. *Advanced Engineering Informatics*, *44*, 101071. https://doi.org/10.1016/j.aei.2020.101071

Leiden, K., & Best, B. (2005). *A cross-model comparison of human performance modeling tools applied to aviation safety*. Micro Analysis & Design, Inc.

Li, L., Das, S., John Hansman, R., Palacios, R., & Srivastava, A. N. (2015). Analysis of flight data using clustering techniques for detecting abnormal operations. *Journal of Aerospace Information Systems*, *12*(9), 587–598. https://doi.org/10.2514/1.I010329

Lüdtke, A., Javaux, D., & Consortium, T. H. (2011). The human project: Model-based analysis of human errors during aircraft cockpit system design. In P. Cacciabue, M. Hjälmdahl, A. Luedtke, & C. Riccioli (Eds.), *Human modelling in assisted transportation* (pp. 79–95). Springer. https://doi.org/10.1007/978-88-470-1821-1_9

Lüdtke, A., Osterloh, J.-P., Mioch, T., Rister, F., & Looije, R. (2010). Cognitive modelling of pilot errors and error recovery in flight management tasks. In P. Palanque, J. Vanderdonckt, & M. Winckler (Eds.), *Human error, safety and systems development: Lecture notes in computer science* (pp. 54–67, Vol. 5962). Springer. https://doi.org/10.1007/978-3-642-11750-3_5

Lüdtke, A., Weber, L., Osterloh, J.-P., & Wortelen, B. (2009). Modeling pilot and driver behavior for human error simulation. *International Conference on Digital Human Modeling*, 403–412. https://doi.org/10.1007/978-3-642-02809-0_43

Masiulionis, T., Miniotas, D., Rudinskas, D., Kikutis, R., & Masiulionienė, G. (2017). Application of automated trajectory design for pilot flight assessment during route flights. *Aviation*, *21*(3), 75–82. https://doi.org/10.3846/16487788.2017.1379440

Memarzadeh, M., Matthews, B., & Avrekh, I. (2020). Unsupervised anomaly detection in flight data using convolutional variational auto-encoder. *Aerospace*, *7*(8), 115. https://doi.org/10.3390/aerospace7080115

Palladino, A., Duff, M., Bock, A., Parsons, T., Arantes, R., Chartier, B., Weir, C., & Moore, K. (2021). Ai-augmented human performance evaluation for automated training decision support. *International Conference on Intelligent Human Systems Integration*, 469–475. https://doi.org/10.1007/978-3-030-68017-6_70

Rehman, U., Cao, S., & MacGregor, C. (2019). Using an integrated cognitive architecture to model the effect of environmental complexity on drivers' situation awareness. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *63*(1), 812–816. https://doi.org/10.1177/1071181319631313

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors*, *48*(2), 362–380. https://doi.org/10.1518/001872006777724417

Salvucci, D. D., & Gray, R. (2004). A two-point visual control model of steering. *Perception*, *33*(10), 1233–1248. https://doi.org/10.1068/p5343

Sarter, N. B., Mumaw, R. J., & Wickens, C. D. (2017). Pilots' monitoring strategies and performance on automated flight decks: An empirical study combining behavioral and eye-tracking data. In R. Key Dismukes (Ed.), *Human error in aviation* (pp. 205–215). Routledge. https://doi.org/10.4324/9781315092898-12

Schoelles, M. J., & Gray, W. D. (2011). Cognitive modeling as a tool for improving runway safety. *16th International Symposium on Aviation Psychology*, 541–546. https://corescholar.libraries.wright.edu/isap_2011/24

Sebok, A., Wickens, C., & Sargent, R. (2013). Using meta-analyses results and data gathering to support human performance model development. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *57*(1), 783–787. https://doi.org/10.1177/1541931213571171

Taatgen, N., & Anderson, J. R. (2010). The past, present, and future of cognitive architectures. *Topics in Cognitive Science*, *2*(4), 693–704. https://doi.org/10.1111/j.1756-8765.2009.01063.x

Verma, S. A., Corker, K., & Jadhav, A. (2003). An approach to modeling error in air-midas using contextual control model. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *47*(1), 21–25. https://doi.org/10.1177/154193120304700105

Xu, R., & Cao, S. (2021). Modeling pilot flight performance in a cognitive architecture: Model demonstration. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *65*(1), 1254–1258. https://doi.org/10.1177/1071181321651008

Zhao, W., Li, L., Alam, S., & Wang, Y. (2021). An incremental clustering method for anomaly detection in flight data. *Transportation Research Part C: Emerging Technologies*, *132*, 103406. https://doi.org/10.1016/j.trc.2021.103406

Tables

Table 1

Average Human Time Duration for Pre-Flight Steps

Checklist Item	Average Time Duration (s)			
	Reading	Checking	Acting	Subtotal
Ignition Switch	1.88	1.88	N/A	3.76
Avionics	0.60	1.06	N/A	1.66
Master Switch	1.07	1.00	1.04	3.11
Fuel Level	1.31	2.62	N/A	3.93
Flaps	1.32	1.49	1.25	4.06
Throttle	1.78	1.34	1.91	5.03
Mixture	1.26	0.94	N/A	2.20
Beacon Light	1.57	1.37	1.14	4.08
Auxiliary Fuel Pump	1.76	1.59	1.16	4.51
Rich Mixture, wait for 5 seconds and then lean Mixture	1.75	1.19	6.47	9.41
Auxiliary Fuel Pump	0.86	0.84	0.74	2.44
Ignition Switch, Mixture, then Ignition Switch check	1.74	5.38	3.34	10.49
Oil Pressure	1.85	3.87	N/A	5.72
Avionics	1.43	1.23	1.10	3.76
Total Time	64.13			

Note: N/A means not applicable because in these cases, no action is needed.

Table 2

Average Human Flight Performance Measures for the Take-Off Task

Measure	Airspeed (knots)	Pitch (degree)	Roll (degree)	Heading (degree)	Total Time (s)
Mean	72.4	9.1	0.3	74.5	231.9
SD	3.1	0.7	0.5	3.8	7.8

Table 3

Parameters Estimated in the Pre-Flight Model

Parameters	Value (s)	Description
imaginal-delay	1.0	Duration needed to form a mental representation.
execution-duration	1.0	Duration needed to move the hand and press a button or switch.
finish-duration	1.0	Duration needed to move the hand back after pressing a button or switch.

Table 4

Average Model Time Duration for Pre-Flight Items

Checklist Item	Average Time Duration (s)			
	Reading	Checking	Acting	Subtotal
Ignition Switch	1.65	1.70	N/A	3.35
Avionics	1.65	1.70	N/A	3.35
Master Switch	1.65	1.65	1.15	4.45
Fuel Level	1.75	1.55	N/A	3.30
Flaps	1.65	1.70	1.20	4.55
Throttle	1.65	1.65	1.15	4.45
Mixture	1.70	1.65	N/A	3.35
Beacon Light	1.70	1.60	1.15	4.45
Auxiliary Fuel Pump	1.70	1.60	1.15	4.45
Rich Mixture, wait for 5 seconds and then lean Mixture	1.65	1.49	7.60	10.74
Auxiliary Fuel Pump	1.65	1.65	1.15	4.45
Ignition Switch, Mixture, then Ignition Switch check	1.70	4.85	4.85	11.40
Oil Pressure	1.65	1.55	N/A	3.20
Avionics	1.65	1.55	1.15	4.35
Total Time	69.84			

Note: N/A means not applicable because in these cases, no action is needed.

Table 5

Average Human and Model Flight Performance Measures for the Take-Off Task

Measure	Airspeed (knots)	Pitch (degree)	Roll (degree)	Heading (degree)	Total Time (s)
Human Mean	72.4	9.1	0.3	74.5	231.9
Model Mean	74.9	7.9	0.0	73.9	241.2
RMSE	2.5	1.2	0.3	0.6	9.3
MAPE	3.5%	13.2%	100%[a]	0.8%	4.0%

Note [a]: this value is an artifact as the average roll values from both human and model were very close to the target value zero, so it should not be regarded as a sign of poor model fitness.

Figures

Figure 1

Devices Used in Human Experiments

Figure 2

Takeoff And Climb Total Time Distribution from All Participants in Task 2

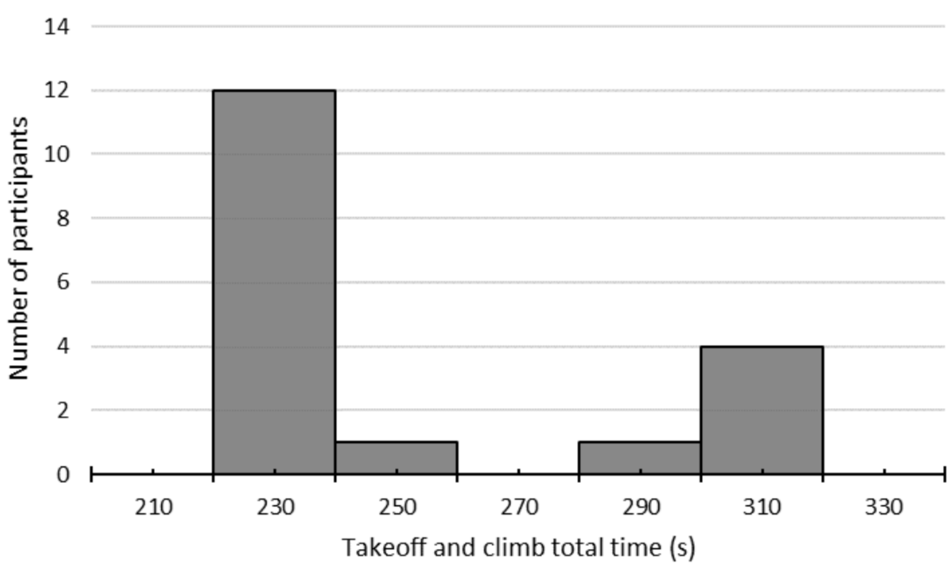
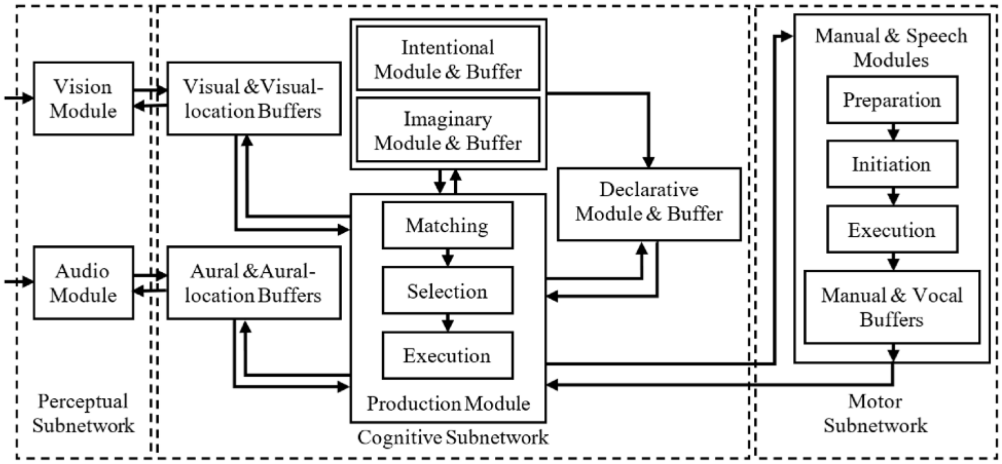


Figure 3

Structure of QN-ACTR



Note. Adapted from "Queueing network-adaptive control of thought rational (QN-ACTR): An integrated cognitive architecture for modelling complex cognitive and multi-task performance," by S. Cao, & Y. Liu, Y., 2013b. International Journal of Human Factors Modelling and Simulation 55, 4(1), 63–86. (https://doi.org/10.1504/IJHFMS.2013.055790).

Figure 4

High-Level Strategy Used in the General Pilot Model

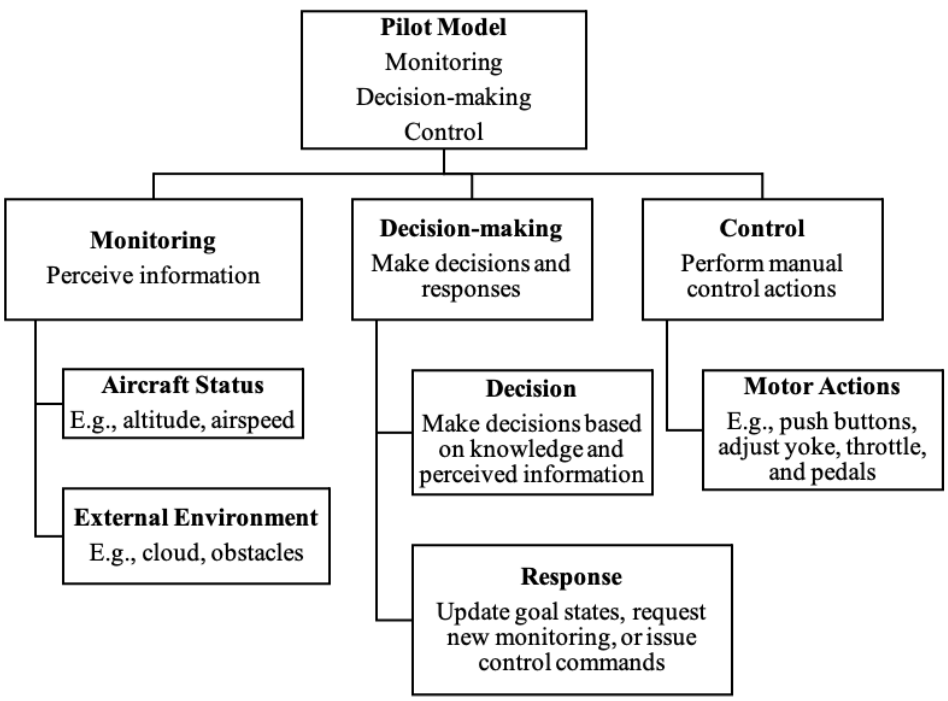
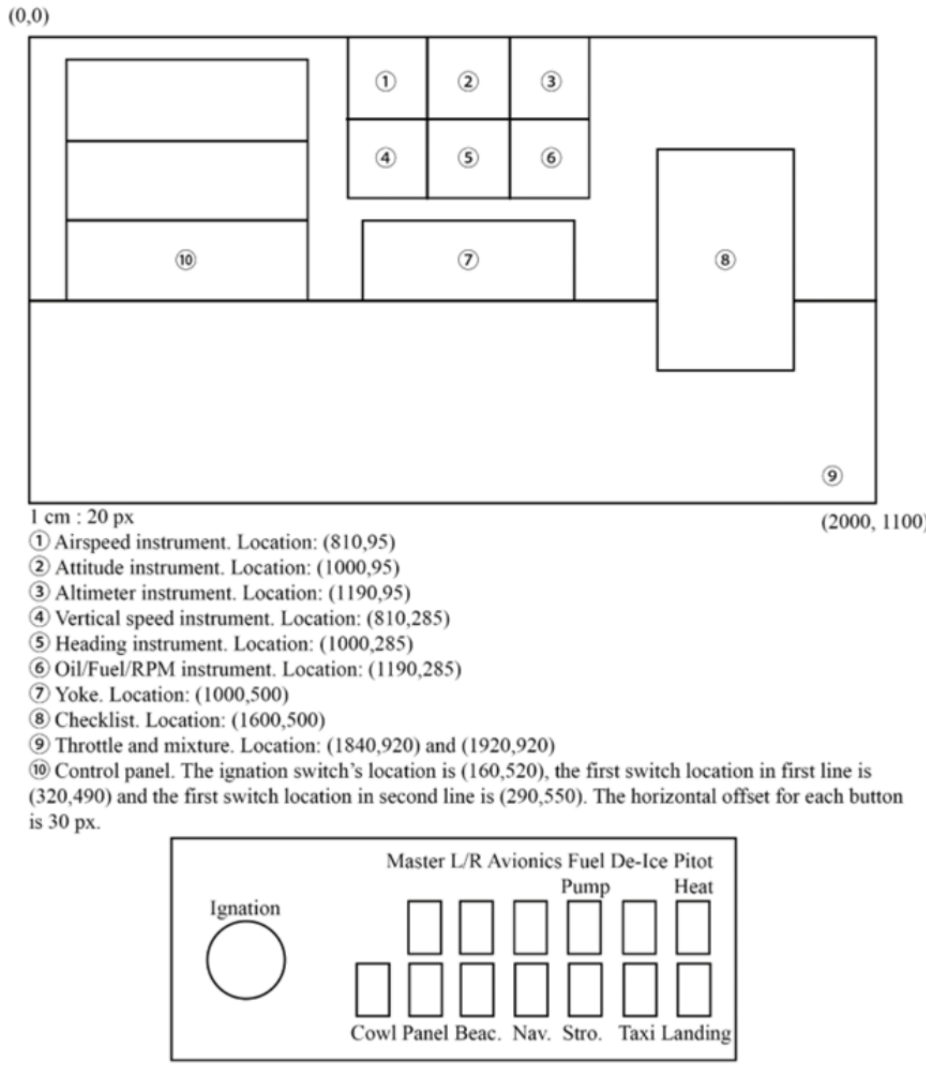


Figure 5

The Control Panel Displayed in Two-Dimensional Coordinate System



A C172SP Checklist Used for Experiments

Appendix A

C172SP Checklist Used for Experiments

C172SP CHECKLIST

FOR EXPERIMENT PURPOSE ONLY. DO NOT USE IN REAL LIFE!

STAGE 1

PRE-FLIGHT

Ignition Switch…………………………………..OFF

Avionics…………………………………………OFF

Master Switch…………………………………..………ON

Fuel Level……………………………..……CHECKED

Flaps…………………………………………………UP

BEFORE START/STARTING/AFTER START

Throttle………………………………..….Open ¼ IN

Mixture…………………………………….CUTOFF

Beacon Light…………………………………..……ON

Auxiliary Fuel Pump……………………..……….ON

Mixture……………………….…FULL 5 SEC/IDLE

Auxiliary Fuel Pump…………………………….OFF

Ignition Switch…………………...………..START

Mixture…………………………………ADVANCE

Oil Pressure…………………...………..CHECK

Avionics……………………………..……………ON

STAGE 2

TAKEOFF

Throttle…………………….………………….FULL

Mixture…………………….…………………..RICH

Lift-off Speed……………….………..……55 KNOT

Initial Climb Speed……….…………70-80 KNOT

CLIMB

Power…………………………………..……FULL

Fuel Level/Temp…………………..…….CHECKED

Engine Instruments………………………………CHECKED

CRUISE

Power……………………………………….ADJUST

Recommended: 75%

B Production Rules and Detailed Descriptions

Overall, the entire procedure is divided into multiple steps, with each step representing processes for each checklist item. Some production rules can be reused in all the steps, so these rules do not need to check the step number in their condition part. Other productions rules are specific to each step, so they need to match the specific step number in their condition part. The model starts from Step 1, Phase 1. For pre-flight part, there is a total of 14 steps, and each step has up to 7 phases; for take-off part, there are 3 phases and they run in a continuous loop.

Table 6

Production Rules and Detailed Descriptions for the Pre-Flight Task (Task 1)

Production Rule	Description
visually-attend-checklist	IF goal step is any step, goal phase is 1, the visual module is free, and the imaginal module is free, THEN set target visual location to the checklist, and change goal phase to 2. *The model simulates pilots setting their gaze to the next checklist item.*
visually-encode-checklist-item	IF goal step is any step, goal phase is 2, a visual location is set, and the visual module is free, THEN move visual attention to the visual location to encode the information and change goal phase to 3. *The model simulates pilots reading the checklist item.*
form-checklist-item-representation-X	IF goal step is X, goal phase is 3, the visual module has encoded the checklist item information, and the imaginal module is free, THEN create a chunk in the imaginal buffer representing the meaning of the checklist item and change goal phase to 4. *The model simulates pilots' understanding from what they read from the checklist item #X. A unique production rule is made for each step. Here X refers to the specific step number such as 1, 2, ..., 14.*
visually-attend-aircraft-component-X	IF goal step is X, goal phase is 4, and the imaginal module has prepared the meaning of the checklist item, THEN set target visual location to the specific aircraft component related to this checklist item and change goal phase to 5. *The model simulates pilots setting their gaze to the aircraft component related to the checklist item.*
visually-encode-checklist-item	IF goal phase is 5, a visual location is set, and the visual module is free, THEN move visual attention to the visual location to encode the information and change goal phase to 6. *The model simulates pilots reading the status of the aircraft component.*
form-aircraft-component-status-representation-X	IF goal step is X, goal phase is 6, the visual module has encoded the information, and the imaginal module is free, THEN create a chunk in the imaginal buffer representing the meaning of the aircraft component status and change goal phase to 7. *The model simulates pilots' understanding from what they read from the aircraft component.*
take-control-action-X	IF goal step is X, goal phase is 7, the imaginal module has prepared the meaning of the aircraft component status, action is needed, and the motor module is free, THEN issue the corresponding control command following the requirement of each checklist item (implemented by calling operation x-plane-button-action or x-plane-continuous-action with corresponding parameters), and change goal step to X+1, goal phase to 1. *The model simulates pilots' control actions for each checklist item, such as turning Master Switch on and setting Mixture to Full for 5 seconds.*
no-action-needed	IF goal step is X, goal phase is 7, the imaginal module has prepared the meaning of the aircraft component status, and no action is needed, THEN change goal step to X+1, goal phase to 1. *The model simulates pilots having no need for any action in this step. For example, Fuel Level is checked to be fine. Move on to the next checklist item.*
checklist-done	IF goal step reaches 15, THEN the checklist is done.

Table 7

Production Rules and Detailed Descriptions for the Take-Off Task (Task 2)

Production Rule	Description
pilot-control-attend-outside	IF goal step is 1, THEN set target visual location to the outside environment and change goal step to 2. *The model simulates pilots setting their gaze to the outside environment.*
pilot-control-perceive-attitude	IF goal step is 2, and a visual location is set, THEN move visual attention to the visual location to encode the information and change goal step to 3. *The model simulates pilots perceiving the aircraft's current attitude including pitch, roll, and yaw angles.*
pilot-control-action	IF goal step is 3, the information from the visual module has been encoded, and the manual module is free, THEN perform actions to control yoke push-pull, yoke rotating, and rudder pedal following Equations 1-3. *The model simulates pilots controlling the aircraft's attitude.*