# Comparing Voice with Touch Screen for Controlling the Instructor's Operating Station of a Flight Simulator

Joël Migneault, Jean-Marc Robert, Michel Desmarais, Sylvain Caron

***Abstract*— Flight simulators are expensive devices that airlines use to train their pilots. Currently, the instructor interact with the simulator by using touch screen devices. We analyzed how a voice driven interface can improve the trainer's interaction time efficiency and fluency with the simulator. Real training scenarios were analyzed and twelve representative tasks were chosen for this study. Time comparisons between the voice driven interface and two touch screen interfaces are reported. Twenty voice commands have been derived from the twelve tasks. The analysis of task completion time for touch screen is based on a model-based approach that relieves us from implementing any interfaces, the KLM-GOMS model. Results show an average execution time gain of 33.8% using voice commands compared to touch screen commands. However, even though the majority of commands have faster input time for the voice activated interface, some are faster to enter through the touch screen, which suggests that an interface that allows both types of interaction mode might be best.**

## I. INTRODUCTION

Flight simulators are very expensive. Training sessions in a simulator cost about $700 to $1000 dollars per hour, depending on the aircraft type. At $1000/hour, cutting a single minute of training per hour translates to savings of about $300 000 per simulator, per year. The incentive to make the training process more efficient is thus paramount. Hence, we need to increase net effective instruction time and reduce the total overhead time incurred in the training process.

We study how to improve the efficiency of simulator based pilot training by using a voice driven interface for the instructor. Our approach is to use an analytical approach that does not require interface implementation. That approach is used to compare the voice driven interface with the existing touch screen interfaces. Let us first describe the flight instructor's operating context, review the relevant literature on voice driven interfaces and the analytical approach we used, and finally report the results of our study.

## II. FLIGHT SIMULATOR TRAINING

A flight instructor's goal is to teach the trainee how to operate an aircraft, and to teach the rules and the security practices and guidelines associated with the aircraft type. He must be able to fly an aircraft from the pilot or the copilot seat and also know every flying maneuvers. He has to communicate his knowledge and evaluate his students.

The instructor has his own operating interface from where he controls the training session. He can work on-board (inside) the simulator before or during the training session. The operating station is designed so the instructor can control the simulation while observing the students from his seat. The station usually has three touch screens, two of 15.4 inches and one of 20 inches. One of the screens is installed on a moving arm so the instructor can look at the screen and control the simulation in front of him while looking at the students. Other screens are generally placed one over the other beside the instructor.

The instructor usually writes down his evaluation and observations regarding the students, reads and follows the training scenario, and simultaneously controls the simulation. His hands and eyes are thus shared between many concurrent tasks. A breakdown estimate of the instructor's activities is: 70% observing students and/or visual, 13% interaction with the operating station, 9% instructing students, 7% role-playing, 1% consulting or taking notes.

The current instructor's interface is touch screen based. This type of interaction has some advantages:

- the input device is the same as the output device (the screen) so that the eye-hand coordination is easy for the user;
- the finger is a natural pointing device for the user so that training is minimal;
- all commands that are available are directly shown on the screen so that one does not have to recall them;

However, touch screens also introduce a number of disadvantages [**?**]:

- the user has to be within an arm-length of the screen to be able to use it: this greatly reduces his mobility in the operating station;
- possibility of fatigue due to the arm position when using the touch screen over a long period;
- the user's arm and hand hides the screen when the user presses a command: this can cause errors and is annoying for the user;
- the size of the finger tip imposes the size of the active area around each command on the screen, the number of commands that can be shown at a time on the screen is limited;

The ability to interact with the simulator over voice recognition opens a new channel of communication. Voice

This work was supported by CAE

Joël Migneault and Jean-Marc Robert are with the Department of Mathematics and Industrial Engineering, Ecole Polytechnique de Montral, Montreal, QC, Canada. (email: joel.migneault@polymtl.ca, jean-marc.robert@polymtl.ca)

Michel Desmarais is with the Department of Computer Engineering, Ecole Polytechnique de Montral, Montreal, QC, Canada. (email: michel.desmarais@polymtl.ca)

Sylvain Caron is with CAE, Montreal, QC, Canada. (email: sylcaron@cae.com)

is also a natural technique for giving commands so that no training is required and space on the screen to display the commands available in the system is no longer a constraint. It give mobility to the user, frees the instructor from keeping the eyes focused on the simulator and, instead, allow better attention to the trainee's behavior and better support for the multitasking nature of the job.

## III. Previous Studies on Speech Recognition

The advantages and shortcomings of speech recognition technology has been the topic of numerous studies in the last few decades. We review the general issues before focusing on the more specific work about speech recognition for flight simulator.

### A. General issues

Weinstein conducted a study in 1995 [**?**] and revealed, after investigating many military and governmental organizations in the United States, that many opportunities were available for voice recognition systems. Even with high noise and stressful user environment, limited grammar and vocabulary in the military domain bring good opportunities for voice applications.

A study done by Draper et al. [**?**] compared manual interaction and voice recognition to control an unmanned aerial vehicle. They showed that voice significantly lowers the number of steps and the overall time to accomplish particular tasks. For a set of normal situations, non critical alerts, critical alerts and information querying, voice reduced time by about 40%. Users also preferred voice interaction over manual interaction on a subjective basis.

Vidulich, Nelson and Bolia attempted in 2006 to verify Draper and his colleagues' results for an Airborne Warning and Control System (AWACS) where operators had to perform simulated battle management command and control tasks [**?**]. Their results revealed that execution time of several tasks were reduced and they showed significant evidence that operators naturally made use of both speech and manual controls together to optimize their performance.

A similar study is mentioned by Williamson et al. in [**?**] and occurred in a military Air Operation Center (AOC) on a system responsible for air tasking orders generation. A first prototype was used to navigate through the existing graphical interface and to input values. Results showed an increase in speed of data entry of 10.6%. The second prototype that enabled multiple events in a single voice command showed even greater speed increase. This research demonstrated the ability of voice recognition systems to increase performance of an air mission planning module.

Studies have compared voice interaction with touch interaction in various domains. Tsimhoni, Smith and Green [**?**] compared these two modes of interaction during car driving simulation. The word-based speech recognition interaction yielded the shortest total task time and also the most favorable results, whereas touch screen interaction brought performance degradation of the vehicle control. Error recovery in this context increased time by 71% for speech recognition and by 23% for touch screen.

Finally, others have found that voice was clearly superior to keyboard for command and control applications when the user is in a situation of high cognitive load or while doing non-linear, concurrent or complex tasks [**?**], [**?**].

### B. Speech Recognition Benefits for Flight Simulators

The integration of speech recognition technology into a flight simulator context is not a new idea. Fox and Weaver introduced this idea twenty years ago [**?**]. Their goal was to decrease the duration and complexity of the interaction with the operating station. At that time, interaction was keyboard-based, such that the instructor had to enter a command line number on the keyboard, the new value for the parameter to modify and finally press "Enter".

Fox and Weaver research lead to a list of five criteria needed for a speech recognition system in this context :

1) the voice recognition system must be much easier to learn to use than current system;
2) any extra demands on the instructor's time should be kept to a minimum;
3) grammar and system command syntax should be kept brief and natural;
4) voice-independent system is more desirable than a dependent one;
5) the system should have the ability to recognize isolated words and continuous speech. Recognition prior to sentence completion could also help the instructor to verify that the recognized command was correct. [?]

Fox and Weaver did not have the voice recognition technology available today and faced problems like speech recognition error rate, the difficulty to train the system and being unable to activate all commands using voice. Since the reaction time to a stimulus has been demonstrated to be faster for a vocal response than a manual response [**?**], we can then expect a gain in execution time through voice command compared to touch screen interaction.

In the specific context of flight simulators, the benefits of speech recognition is that it reduces the overhead time required for operations like simulation initialization, planning, and interaction. It allows more time for the instructor to instruct and brief students on assigned tasks.

## IV. Methodology: Predictive Human Performance Models

Our study relies on an analytical approach to compare the voice vs. touch screen interfaces. This approach was pioneered by Card, Moran and Newell (1983) ajouter dans références who introduced the GOMS model. It allows the prediction of task completion time over different interfaces without the need to actually implement the user interface itself.

The GOMS (Goals, Operators, Method, Selection) task analysis method describes the human-computer interaction as a hierarchy of goals, sub-goals and elementary actions.

Using these elementary actions, we can estimate the time required for an expert user to accomplish a certain task without the need of any observation or task recording. Two parts must be considered in an elementary action: preparation time and execution time. This method yields quantitative estimates of the time these actions would take. A more precise version of GOMS method, called KLM-GOMS (Keystroke-Level Model) [**?**] helps analysts to evaluate different graphical interface designs by providing data about user performance using those interfaces. It uses time predictive models of human performances with different input devices at the keystroke level and is the basis of the CogTool analysis.

This analysis method has been used in various research. John et al. [**?**] report real world applications of this method that demonstrate its validity. The *Atomic Components of Thought (ACT-R)* referred in [?] is a complete software architecture that simulates and predicts human performance and cognition. The ACT-R research group in the Department of Psychology at the Carnegie Mellon University has successfully created models about human learning and memory, problem solving and decision making, etc. [**?**] They compared the model results with the results of people doing the same task. Time to perform the task, accuracy in the task and neurological data are cognitive psychology measures taken during this validation process. The software CogTool is based on this ACT-R framework and has been built to provide a tool to make predictive human performance modeling easier [?], [**?**].

## V. ANALYSIS OF TASK COMPLETION TIME

The objective of this analysis is to compare voice vs touch screen task completion time. Since these tasks are tightly associated with the training scenario, we need to gather data about the instructor tasks.

### *A. Scenario-based Investigation*

Some of the tasks done by the instructor with the operating station during this scenario are listed in table I. They are used asthe basis of comparison for this analysis. The number of times the instructor does a task during a three hour session is identified in the right column of this table. Most of the tasks chosen for this analysis are related to simulation parameters initialization. The instructor has to reinitialize the simulation every time the training scenario used by the instructor needs to place the students in a different context. These tasks are some of the most frequent in a training session. However, we note that these tasks constitute only a portion of the training session time as explained in section II.

TABLE I
TASKS DONE DURING A 3 HOUR TRAINING SCENARIO AND VOICE COMMANDS ASSOCIATED WITH EACH TASK

| **TASKS (↪ Voice commands)** | **#** |
|---|---|
| **TASK: SET FUEL QUANTITY IN THE AIRCRAFT**<br>↪ Set fuel weight to fifteen thousand | 2 |
| **TASK: SET REFERENCE AIRFIELD**<br>↪ Set reference airfield to L I R F<br>↪ Set reference runway to sixteen L | 7 |
| **TASK: REPOSITION THE AIRCRAFT AT POSITION**<br>↪ Reposition to takeoff | 7 |
| **TASK: ACTIVATE THE SIMULATION**<br>↪ Unfreeze flight | 5 |
| **TASK: SET CLOUDS**<br>↪ Set cloud cover to scattered<br>↪ Set cloud visibility to zero point five<br>↪ Set cloud top to ten thousand<br>↪ Set cloud base to two hundred | 4 |
| **TASK: SET WIND**<br>↪ Set wind direction to one eight zero<br>↪ Set wind speed to forty five<br>↪ Set air temperature to twenty | 7 |
| **TASK: SET RUNWAY CONDITIONS**<br>↪ Set runway roughness to three<br>↪ Set runway contaminant to Dry | 1 |
| **TASK: SET VISUAL TO A TIME OF DAY**<br>↪ Set visual to night | 2 |
| **TASK: ACTIVATE A MALFUNCTION**<br>↪ Set malfunction right engine flameout at twenty feet | 7 |
| **TASK: SET SIMULATION SPEED**<br>↪ Set slew forward at three | 2 |
| **TASK: RESET SIMULATION**<br>↪ Reset all systems<br>↪ Reset all temperatures | 3 |
| **TASK: DEACTIVATE ALL MALFUNCTIONS**<br>↪ Clear malfunctions | 2 |

### *B. Touch Screen Analysis*

Using the scenario previously described, we gather the task completion time for an expert user over two different versions of the operating station's graphical interface using touch screen device (see figure 1). Since data collection in real training situation is quite expensive and difficult to obtain, we use the KLM-GOMS task analysis methodology described earlier through the CogTool [**?**] software to estimate these execution time. The process of gathering execution time data using this tool requires adding screen shots of the different designs we want to analyze, then we need to identify controls, the actions these controls generate and finally record the sequence of events that constitutes the tasks we want to analyze. Each task needs a particular sequence of interaction with the operating station. A time of 1.35 seconds is added before every interaction with the touch screen to consider the mental operator as defined by Card, Moran and Newell and used in the ACT-R human performance model under CogTool. This time considers unobservable processes such as remembering commands, visual localization of elements, etc.

We compare two versions of the operating station's graphical interface. The two different versions are used to validate the touch screen interaction mode at the operating station. The use of two different interfaces allow a better estimate of the impact of interface design on task completion time, as task could vary considerably between different interface designs. Thus, using two designs provide higher reliability. Both interfaces use touch screen input devices inside the simulator. The first version (figure 1(a)) is actually used in most commercial airlines' training simulators, while the other (figure 1(b)) is a newly designed interface built to maximize information available to the instructor at any given

time. Details about these graphical interfaces cannot be provided for confidentiality reasons.

(a) GUI 1

(b) GUI 2

Fig. 1. Two versions of the instructor operating station's graphical interface (with courtesy of CAE)

### C. Voice Analysis

Voice command analysis is done using two methodologies in order to validate the estimation results. We first estimate the time needed by a user to execute tasks using voice. An average voice production rate of 175 words per minute is used as reference based on the listed rates gathered from the various research given in table II. This estimate, based on the number of words contained in each command (see table I), gives us a first value for the possible gain of voice input on manual input. Then, we record six human subjects while they read out loud the same set of commands. A manual analysis of the wave signal of the recordings gives us the time for each command to be uttered by a particular subject, which leads us to an estimate of the average time observed for each command. We then compare these estimates to those computed using the average 175 words per minute from the literature and see if the value is an accurate estimate in this context. We finally compare our estimated voice command execution times with the touch screen execution time for the same overall tasks and draw conclusions about the potential of voice recognition at the instructor's operating station.

The set of voice commands in table I have been derived from the scenario observed, by using action commands and labels used in the graphical interface forms. The value for parameters have been taken from the training scenario. The same values are used during touch screen analysis.

Finally, since the model used to estimate execution time with touch screen interaction considered the mental operator before each action, we also add 1.35 seconds before each command to make our comparison more accurate.

TABLE II

SPEECH RATES OBTAINED FROM VARIOUS SOURCES

| Rate | Reference |
|---|---|
| 150 to 200 words per minute | Newell et al. (2003) [**?**] |
| 111 to 291 words per minute | Yuan et al. (2006) [**?**] |
| 223 words per minute (for man) | Liberman (2006) [**?**] |
| 125 to 150 words per minute | Fulford (1992) [**?**] |
| 150 to 250 words per minute | Rossi et al. (1981) [**?**] |
| 150 words per minute | Minker and Bennacef (2004) [**?**] |
| 300 syllables per minute | Wood (1973) [**?**] |

## VI. RESULTS

### A. Touch Screen

Using print screens of both versions of the graphical interface into the CogTool software, we get the task execution times shown in table III. These results show that the first version is faster than the second version. Fitts' law can explain this difference since the second interface has smaller buttons than the first one. These buttons are at the minimum limit described in graphical interface standards for touch screen devices [**?**].

The fastest tasks are to activate the simulation and deactivate malfunctions, taking an average of 1.9 seconds, since the buttons are directly accessible on both interfaces. The longest task, 30.1 seconds on the second version, is to set cloud coverage in the simulation.

### B. Voice

Voice interaction has been estimated at 175 words per minute according to previous study reported in the literature. Six subjects have been recorded, five men and one woman from 25 to 40 years old, and we observed an average production rate of 156 words per minute. Both methods are compared in table III. Similar results are obtained with an average time of 3.1 seconds using the rate from theory and 3.3 seconds for the empirical data. These results consider the 1.35 second for mental preparation before saying a command. The slowest spoken command took 4.5 seconds to utter 8 words while the fastest took 2.4 seconds for 2 words.

### C. Comparison Results

Using the results from touch screen and voice analysis previously presented, we have compared these two interaction techniques for the same tasks. Figure 2 compares results from the average voice commands execution time and both versions of the interface using touch screens. The graph shows a significant difference between both interactions and a faster execution time using voice. The figure also shows a greater difference between voice and touch screen interaction

TABLE III

TASK EXECUTION TIME IN SECONDS ESTIMATED WITH THE COGTOOL SOFTWARE FOR THE TWO VERSIONS OF THE GRAPHICAL INTERFACE (GUI 1, GUI 2) AND VOICE PRODUCTION TIME USING THEORETICAL DATA (VOICE 1) AND EMPIRICAL DATA (VOICE 2)

| TASKS | GUI 1 | GUI 2 | GUI MEAN | VOICE 1 | VOICE 2 | VOICE MEAN | MEAN DIFF. GUI-VOICE |
|---|---|---|---|---|---|---|---|
| Set fuel quantity in the aircraft | 10.2 | 10.5 | 10.3 | 3.4 | 3.4 | 3.4 | 6.9 |
| Set reference airfield | 7.0 | 10.3 | 8.7 | 7.5 | 6.4 | 6.9 | 1.7 |
| Reposition the aircraft at position | 3.7 | 6.0 | 4.8 | 2.4 | 2.7 | 2.6 | 2.3 |
| Activate the simulation | 1.9 | 1.8 | 1.9 | 2.0 | 2.4 | 2.2 | -0.3 |
| Set clouds | 27.4 | 30.1 | 28.7 | 13.6 | 9.6 | 11.6 | 17.1 |
| Set wind | 18.8 | 10.7 | 14.8 | 10.2 | 7.5 | 8.9 | 5.9 |
| Set runway conditions | 7.3 | 13.0 | 10.1 | 6.1 | 5.6 | 5.8 | 4.3 |
| Set visual to a time of day | 3.8 | 3.9 | 3.8 | 2.7 | 2.6 | 2.7 | 1.2 |
| Activate a malfunction | 12.4 | 13.7 | 13.1 | 4.1 | 4.5 | 4.3 | 8.8 |
| Set simulation speed | 7.3 | 7.8 | 7.5 | 3.1 | 3.3 | 3.2 | 4.3 |
| Reset simulation | 5.3 | 5.4 | 5.4 | 4.8 | 4.1 | 4.4 | 0.9 |
| Deactivate all malfunctions | 1.8 | 1.9 | 1.9 | 2.0 | 2.6 | 2.3 | -0.4 |
| | 8.9 | 9.6 | 9.2 | 5.2 | 4.6 | 4.9 | 4.4 |

when the number of words and number of subtasks increase. However, the fastest manipulations using the touch screen are done faster than the corresponding voice command, even though these commands have a small number of words. This is mainly a result of the button being always accessible on the interface, which means that the user has only one mental preparation and one press to execute in order to finish the task.

Fig. 2. Mean voice commands execution time compared to mean touch screen tasks execution time ordered according to the number of words/sentences used to accomplish a task

Time gain percentage is reported in figure 3. The average gain in execution time is 33.8% using voice. The highest gain observed is 67% for setting a malfunction. Negative gains show that some tasks are faster using direct manipulation. Considering the number of times each task would be executed during the scenario described previously, the instructor would gain roughly 4 minutes of time using a subset of 20 voice commands for the 12 tasks analysed. We estimate that more than 200 commands are possible and integrating more of them would increase this gain of time.

Considering error recovery times taken in [?], there would be no gain for some tasks while using voice. However, voice would be faster for particular tasks and also let the instructor work in a more natural way.

## VII. CONCLUSIONS AND FUTURE WORK

We demonstrate the use of an analytical approach to compare voice with touch screen at an instructor operating station. Two versions of a graphical interface for this station have been analyzed using KLM-GOMS model through the CogTool software. Results show that we can decrease task execution time by up to 67% by using voice. However, some tasks are performed faster using the touch screens when only one button needs to be pushed and when it is directly accessible from the main interface screen. Voice command is surely a promising interaction mode for this context.

However, this analysis does not consider speech recognition errors, nor error recovery time with both interaction modes. Our goal was to obtain a gross estimate of the gain that voice would produce over touch screen in an ideal situation. Also, no command confirmation has been considered while interacting vocally since errors are not critical in this training situation. We also think that an average gain of 33.8% would give the user time to recover from false recognition and still gain task execution time overall. Moreover, using the GOMS task analysis method does not consider nonprocedural aspects of usability such as readability or memorability which can have a significant impact the task execution time. However, voice command interaction also has memorability issues and more usability tests would be needed in both cases.

Fig. 3. Percentage gain of voice commands over touch screen interaction using average task execution times

Given the encouraging results so far, we now plan to build a prototype of the voice recognition interface and test it in a real flight simulator. Noise is a significant factor that will need to be addressed since it could have an impact on the speech recognition error rate. However, previous studies have shown that voice recognition was possible in a real cockpit environment with positive results. We also plan Wizard-of-Oz experiments to identify the correct vocabulary and grammar to use in the voice activated interface. This aspect must be carefully investigated since we don't want to overload the instructor's mental workload or increase the operating station learning curve. This study is a first step towards our goal of finding the best fit of human factors for this particular human-computer interaction.

## VIII. ACKNOWLEDGMENTS

The authors gratefully acknowledge the technical and knowledge contribution of CAE, as the human factor engineers Sebastien Malo and Mireille Audet for their professional support.