



ARL-TR-9012 • AUG 2020



Development and Usability Assessment of the Real-time Event, Flow, and Coordination Tool (REFLECT)

by Anthony L Baker, Ralph W Brewer, and Kristin E Schaefer

Approved for public release; distribution is unlimited.

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.



Development and Usability Assessment of the Real-time Event, Flow, and Coordination Tool (REFLECT)

Anthony L Baker
Oak Ridge Associated Universities

Ralph W Brewer
Vehicle Technology Directorate, CCDC Army Research Laboratory

Kristin E Schaefer
Human Research and Engineering Directorate, CCDC Army Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) August 2020		2. REPORT TYPE Technical Report		3. DATES COVERED (From - To) August 2019–March 2020	
4. TITLE AND SUBTITLE Development and Usability Assessment of the Real-time Event, Flow, and Coordination Tool (REFLECT)				5a. CONTRACT NUMBER W911NF-18-2-0230	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Anthony L Baker, Ralph W Brewer, and Kristin E Schaefer				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) CCDC Army Research Laboratory ATTN: FCDD-RLH-FD Aberdeen Proving Ground, MD 21005-5425				8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-9012	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES ORCID ID: Kristin E. Schaefer, 0000-0002-1342-3446					
14. ABSTRACT This report details the development and initial usability assessment of the Real-time Event, Flow, and Coordination Tool (REFLECT), which allows a user to capture team communication flow in real time. This is critical for advancing US Army initiatives in human–autonomy teaming because communication flow affects, and is affected by, intelligent systems interactions, team structures, and tasking demands associated with the novel metric of team trust and cohesion. In our initial usability assessment, participants used REFLECT to capture team communication flow using audio recordings from the US Army Wingman Joint Capabilities Technology Demonstration, where a five-person crew engaged in mounted machine-gun live-fire exercises involving a manned–unmanned team. After using REFLECT, participants completed questionnaires and interviews regarding its design and use. Participants achieved approximately 70% accuracy in correctly identifying speakers and recipients of communication exchanges with an average delay of about 2 s. Participant feedback suggests that the tool is easy to use but also provides directed insight into areas for improvement.					
15. SUBJECT TERMS communication, human–autonomy teams, teamwork, intelligent agents, communication analysis					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 39	19a. NAME OF RESPONSIBLE PERSON Anthony L Baker
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			19b. TELEPHONE NUMBER (Include area code) 410-278-5396

Contents

List of Figures	v
List of Tables	vi
Acknowledgments	vii
Summary	viii
1. Introduction: Real-time Event, Flow, and Coordination Tool (REFLECT)	1
1.1 Rationale for Development of REFLECT	1
1.2 REFLECT Development	2
1.2.1 Initial Design	2
1.2.2 REFLECT Features and Functions	3
1.2.3 Data Output	8
2. Methods: Usability Assessment of REFLECT	10
2.1 Participants	10
2.2 Crew Audio Recording	11
2.3 Training	11
2.4 Usability Task	12
2.5 Usability Questions	12
2.6 Data Preparation	13
3. Results	13
3.1 REFLECT Performance	14
3.2 Usability Feedback	16
3.3 Usability Questions	17
4. Discussion	17
4.1 Training	18
4.1.1 Familiarity with Voices	18

4.1.2	Familiarity with Task Context	18
4.2	Unitizing Communication Events	19
5.	REFLECT Updated Development	20
5.1	Change Hot Keys	20
5.2	Undo Speaker Action	20
5.3	Custom Audio Selection	21
5.4	Future Tool Development	21
6.	Conclusions	22
7.	References	24
	List of Symbols, Abbreviations, and Acronyms	27
	Distribution List	28

List of Figures

Fig. 1	GUI is separated into four zones: 1) audio, 2) input, 3) events, and 4) editing	3
Fig. 2	REFLECT setup screen. Note the setup prompts in the input pane on the left side of the tool as indicated by the blue zone (no. 2) from Fig. 1.....	5
Fig. 3	Create a custom crew setup. Enter new names for each of the crew members and then select the Create Custom Buttons button.....	5
Fig. 4	REFLECT main window after a team has been configured. The input pane (left) is used for logging communication events. The event pane (center) displays communication events as they are collected. The editing pane (right) is used for appending comments or other text to communication events.	6
Fig. 5	Example run using REFLECT in support of a Wingman exercise. The left pane shows an example of a sender–receiver message. The center pane displays the sender, receiver, and a timestamp for each communication event. The right pane allows the user to adapt information collected in the center pane.	7
Fig. 6	Communication events that have been transcribed using the right pane. Note each event in the middle pane now has text appended after the timestamp.....	8
Fig. 7	Example data file where data were exported from REFLECT as a .csv file and opened in Microsoft Excel. Column A indicates the sender. Column B indicates the receiver. Column C contains timestamps down to the second, though a future version will account for milliseconds as well. Column D contains any text information that was appended to each communication event.....	9
Fig. 8	Communication flow diagrams for Army and Marine crews during human–autonomy gunnery events during target engagement tasks. Thicker lines indicate more communication from that speaker to the indicated recipient.	9
Fig. 9	REFLECT usability questions	13
Fig. 10	Usability ratings provide by participants. Error bars indicate 1 standard deviation.....	17
Fig. 11	Zoomed-in screenshot depicts the method to change hotkeys assigned to each role. User clicks Change Hotkeys and then drags each number onto a role.	20

List of Tables

Table 1	Percent effectiveness of identifying sender–receiver pairs using REFLECT across the two test runs for each participant.....	15
---------	-----------------------------------------------------------------------------------------------------------------------------	----

Acknowledgments

This research was supported in part by an appointment to the US Army Research Postdoctoral Fellowship Program administered by the Oak Ridge Associated Universities through a cooperative agreement with the US Army Combat Capabilities Development Command (CCDC) Army Research Laboratory (ARL). Research was sponsored by the CCDC Army Research Laboratory and accomplished under Cooperative Agreement Number W911NF-18-2-0230.

This work supports the Human–Autonomy Teaming Essential Research Program Project 5 on team trust and team cohesion. The authors thank Dr Kimberly Pollard for providing technical review and critical guidance for this report.

Summary

Communication is critical to understanding teamwork because it provides a window into how teams share information, coordinate toward goals, solve problems, and achieve performance objectives.¹ Future human–autonomy teams will be multidimensional and involve novel teaming structures between humans and intelligent agents. If we are to understand performance and teamwork in these future teams, it will be critical to understand their communication.

This report details the development of a novel method for assessing verbal crew communication flow in real time through the Real-time Event, Flow, and Coordination Tool (REFLECT), developed by the authors of this report. REFLECT is a graphical user interface (GUI) developed in Java that allows users to log a team’s communication based on the speaker (sender) and the intended recipient (receiver) of each message. While the prototype for REFLECT is aimed at assessing verbal messages, the tool is ultimately intended to allow the logging of senders and receivers via any communication method used by teams, especially human–autonomy teams. Features for the automatic logging of senders and receivers are planned for future versions.

Following the description of the tool, results from an initial usability assessment of REFLECT are provided. Participants used REFLECT to assess team communication flow on audio recordings from the US Army Wingman Joint Capabilities Technology Demonstration, where a five-person crew of US Marines engaged in mounted machine-gun live-fire exercises involving a manned–unmanned team. After using REFLECT, participants completed questionnaires and interviews regarding its design and use. Participant feedback suggests the tool is easy to use but areas for improvement exist. For example, users requested the ability to add and reassign hotkeys, add keyboard shortcuts, quickly undo errors, and navigate audio recordings more easily. These findings were used to improve tool features, including the creation of custom hotkeys, an undo option to reverse the last event entry, and the ability to jump to different points in an audio recording using a seek bar.

Participants reported that the most difficult aspect of using REFLECT was keeping up with the rate of spoken communication between team members in an audio recording of human–autonomy team operations. Results suggest that to log communication events quickly and accurately, a REFLECT user should be familiar with team members’ voices and the team’s task context to make efficient judgments

¹Salas E, Sims DE, Burke CS. Is there a “big five” in teamwork? *Small Group Research*. 2005;36(5):555–599.

as to the sender and receiver of each communication event. However, the addition of automated methods for logging senders and receivers, such as via push-to-talk button presses from each team member, will circumvent some of those requirements.

Additional features are planned for implementation. This prototype of REFLECT works by allowing the user to import audio recordings for communication event logging, but an update will allow it to be used for live scenarios as well. REFLECT also currently logs timestamps down to the second, but an update will allow for more-precise millisecond timing of event logs.

The next versions of REFLECT will be tested during upcoming field exercises and Information for Mixed Squads laboratory simulation studies. This work is critical for advancing Army initiatives in human–autonomy teaming (HAT) as outlined by US Army Combat Capabilities Development Command Army Research Laboratory’s HAT Essential Research Program because communication flow affects, and is affected by, intelligent systems interactions, team structures, and tasking demands. It is also important to understanding elements of team trust and cohesion.

1. Introduction: Real-time Event, Flow, and Coordination Tool (REFLECT)

REFLECT was developed by the authors of this report to be able to rapidly and effectively measure the flow of verbal communication in multisized teams, including human–autonomy teams (HATs). Before the development of this tool, this type of analysis was conducted by hand a posteriori, requiring recorded audio interactions and transcribed data. The goal of this tool is to capture data that can be used to map how communication flows within the team, which can provide insights into shared cognitions, team cohesion, and trust. Thus, the communication flow data captured by REFLECT can inform analyses of the relationships between team communication and performance. It is adaptable to teams of various sizes, functions, and capabilities, and can capture aspects of a team’s coordination that reveal insights into other critical constructs such as situation awareness (SA) and team cohesion.

1.1 Rationale for Development of REFLECT

Communication is a fundamental aspect of team performance. Teams communicate to exchange and coordinate information, and the ways that teams communicate have implications for their performance. Aspects of team communication can provide insights into team functioning (Salas et al. 2008), team cognition (Cooke et al. 2013), and overall performance (Marlow et al. 2018). Likewise, the patterns exhibited by teams during communication exchanges can be used to understand how information is flowing within a team.

For example, approaches such as social-network analysis have been used to model and study communication networks by evaluating the relational structures of groups of varying sizes (Butts 2008; Newman 2010). Data from this type of analysis demonstrate the quantity of information transmitted among team members and can reveal trends of communication patterns within a team. For example, network analysis can reveal the extent to which different members in an organization communicate with each other or the interaction patterns of crew members over a given time period.

Other approaches have considered how the flow of communication within a team relates to team performance. For example, Fischer et al. (2007) evaluated the rate of communication in a simulated search-and-rescue task to demonstrate the structural differences in the communication patterns of successful and unsuccessful teams. Findings provided some evidence that successful search-and-rescue teams

have a more equal distribution of communication than unsuccessful teams in which a few team members dominated the conversations.

The design of the REFLECT tool is built on these methods as well as theory and models of team communication, such as the communication models of Shannon (1948) and Berlo (1960), the turn-taking (TT) theory of Sacks et al. (1974), and the meta-analytic model relating team communication and performance of Marlow et al. (2018; for a more in-depth review of these works, see Baker et al. [2019]). This work supports the US Army Combat Capabilities Development Command (CCDC) Army Research Laboratory's (ARL's) Human–Autonomy Teaming (HAT) Essential Research Program by developing a way to measure team communication so that effective measurement and associated metrics of team trust and team cohesion for HATs can be identified.

1.2 REFLECT Development

To maintain overmatch in changing adversarial environments and against emerging threats, future HATs must be characterized by dynamic, naturalistic interactions to perform, adapt, and succeed. Future HAT will be multidimensional and involve novel teaming structures between humans and intelligent agents. Although we do not yet know what these future teams will look like, we will need effective methods of collecting and measuring how they communicate to understand and explain their performance. In the nearer term, these methods of measuring communication will support ongoing research and development of future HAT. To serve these needs, REFLECT is adaptable to teams of various sizes, functions, and capabilities. It is designed to capture the interactions between team members to reveal the unique coordination characteristics (e.g., trust, cohesion, and SA) of those teams.

1.2.1 Initial Design

The REFLECT application is developed in Java because Java programs are portable, meaning that the bytecode can be used on any computer hardware that has a compliant Java Virtual Machine. In addition, Java uses a more rigorous set of semantics that provide powerful libraries that help create graphical user interfaces (GUIs) without spending a lot of time writing code. For the REFLECT application, the Eclipse plugin “WindowBuilder” was used to create the GUI using a “what you see is what you get” visual designer (Wren n.d.).

The latest version of the REFLECT GUI (Fig. 1) is separated into four distinct zones: audio (red), input (blue), events (green), and editing (orange). The audio zone correlates with the `AudioPlayer.java` and `PlayingTimer.java` files, which allow a user to load, play, pause, resume, stop, step forward or step back, and set the time

using the slider. The input zone outlines the instructions for the application, setup for the input window, and the input actions. The SketchPad.java file provides the panel for connecting the communications visually. The events zone outlines each communication input, output, time, and comment. The editing zone allows the user to edit, add, or delete communications in the events zone. The final two zones are brought together as the others were, using the standard components such as text boxes, scroll bars, labels, and more to textually show the communications and allow changes to those communications.

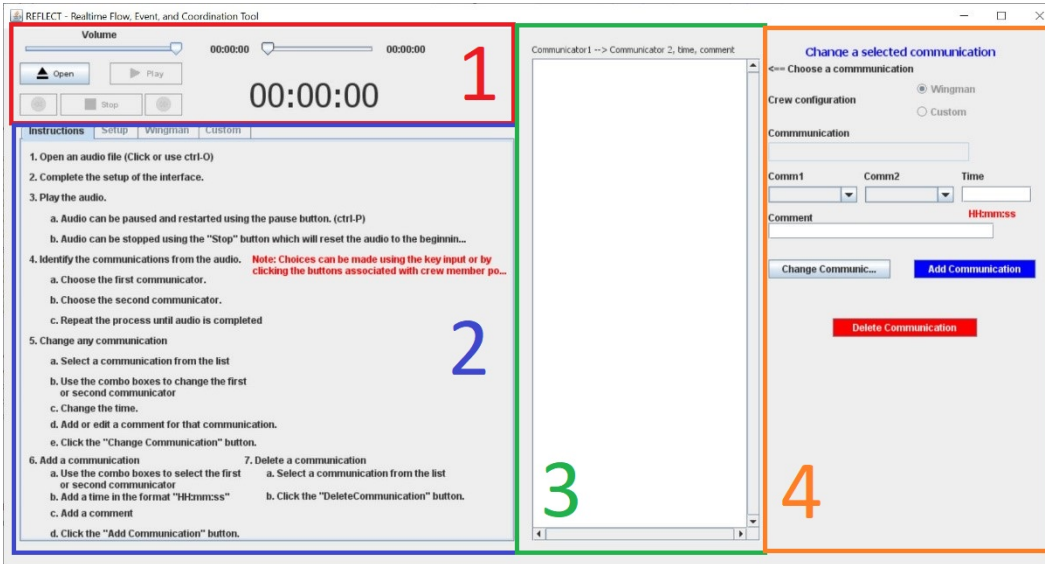


Fig. 1 GUI is separated into four zones: 1) audio, 2) input, 3) events, and 4) editing

1.2.2 REFLECT Features and Functions

Since REFLECT is aimed at capturing a team’s communication patterns, the current prototype of REFLECT requires an observer to log a team’s communication-based sender-to-receiver events. Each communication event is then automatically timestamped based on the time at which the observer enters each communication event.* While our first tests of REFLECT are looking at verbal crew communication, this tool can also be applied to any communication method used by diversified team members (including autonomy) as long as there is a sender and receiver for each message.

Other software tools exist for annotation and event logging. Some examples are ELAN (Wittenburg et al. 2006), which is aimed at audio/video annotation, ANVIL (Kipp 2012), which is designed for video annotation (but can be used on audio

*This behavior will be expanded upon in future versions of REFLECT to improve the accuracy of the generated timestamps (see Section 5). For example, during a live scenario, a future voice-detection feature might produce an automated timestamp when a team member’s speech is detected.

files), and Praat (Boersma 2001), which was designed for phonetics and audio annotation, though its features are customizable and extensible by using scripts. These software packages offer a variety of features designed specifically for accurate, comprehensive transcription of video or audio data and allow for much detail to be gleaned from a given data recording. However, unlike these software packages, REFLECT is purpose-built for rapid, on-the-fly annotation of communication interactions. Its features are focused on supporting a user's ability to observe a team interaction and enter data real-time to characterize who is speaking to whom. Such data can allow for insights into the performance and effectiveness of teams (Schaefer et al. 2019a), and the speed at which these data can be generated is a benefit for applications, such as field research or training scenarios.

The current prototype version of REFLECT is built to be used with an audio recording of team interactions, whereas capabilities are being added that will allow REFLECT to be used during live scenarios. Eventually, when voice- and speech-detection algorithms are sufficiently intelligent, the human-in-the-loop of REFLECT can be removed, allowing the algorithms to handle the logging of communication interactions (see Section 5). The following sections detail the program's features as the user progresses through setup, data collection, and data output.

1.2.2.1 Setup

Upon running the executable jar file for REFLECT, the first step is to open an audio file by clicking on the open button in Zone 1 (Fig. 1). Then the user will be presented with the setup screen (Fig. 2). Here, REFLECT prompts the user to configure the team that will be observed. There are two choices: the first is to use the Wingman crew and the second is to create a custom crew.

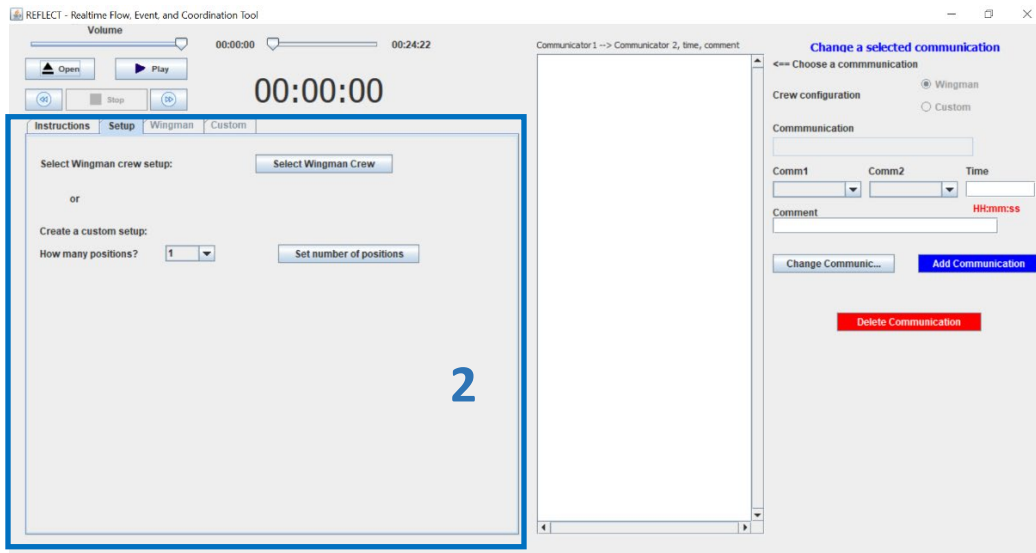


Fig. 2 REFLECT setup screen. Note the setup prompts in the input pane on the left side of the tool as indicated by the blue zone (no. 2) from Fig. 1.

The custom setup allows the user to choose a crew size, which then prompts them to enter unique crew member names (Fig. 3).

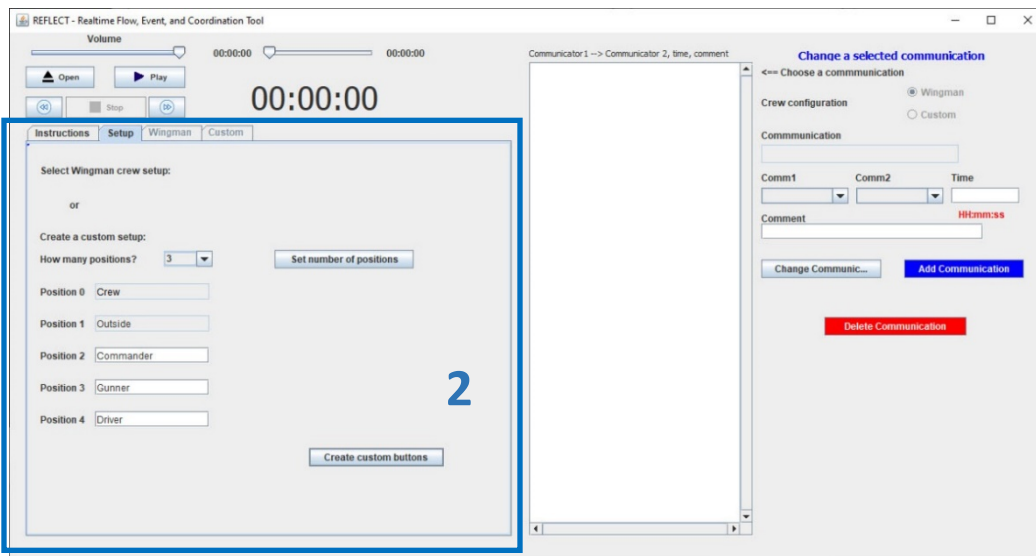


Fig. 3 Create a custom crew setup. Enter new names for each of the crew members and then select the Create Custom Buttons button.

The roles for Crew and Outside are automatically provided. The former is used to indicate when a team member communicates to the team as a whole, while the latter is useful when a team member communicates with someone outside of the team being analyzed. After recording these values and pressing Create Custom Buttons, the next screen will populate with the roles that the user provided. The number of

crew members indicated in REFLECT determines the arrangement of roles on the input pane.

Figure 4 provides an example of a five-person human-only crew from the US Army Wingman program operating a remote unmanned weaponized ground vehicle from a command and control vehicle (Brewer et al. 2018). While this example involves only a human crew, future versions of REFLECT will support communication between humans and/or autonomous agents as long as each communication event has a discernible sender and receiver.

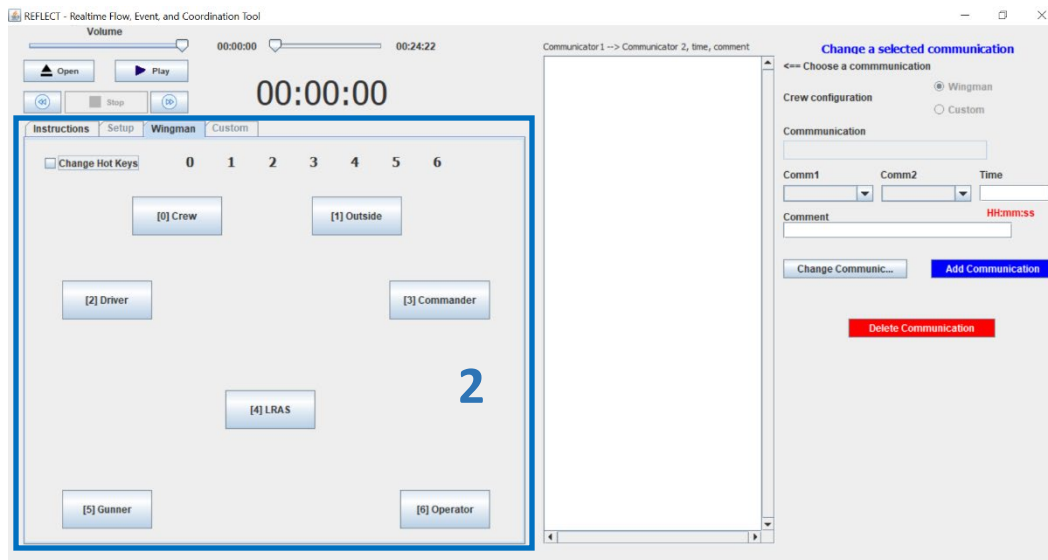


Fig. 4 REFLECT main window after a team has been configured. The input pane (left) is used for logging communication events. The event pane (center) displays communication events as they are collected. The editing pane (right) is used for appending comments or other text to communication events.

1.2.2.2 Collecting Data

Using the event pane (left side of Fig. 5 with icons and arrow), the user can log the source and destination of each verbal message. When the user has opened an audio file and is ready to begin logging communication events, the user can press Play. When a crew member begins speaking, the user clicks that person's role, which turns red. The user then clicks the intended recipient for the team member's communication, which turns blue. This completes one communication event. Each communication event is populated in the event pane in the middle of the user interface, along with the timestamp (in seconds) of the communication event, relative to the position in the audio file. The user continues logging communication events *during* the team's interactions until the scenario is complete.

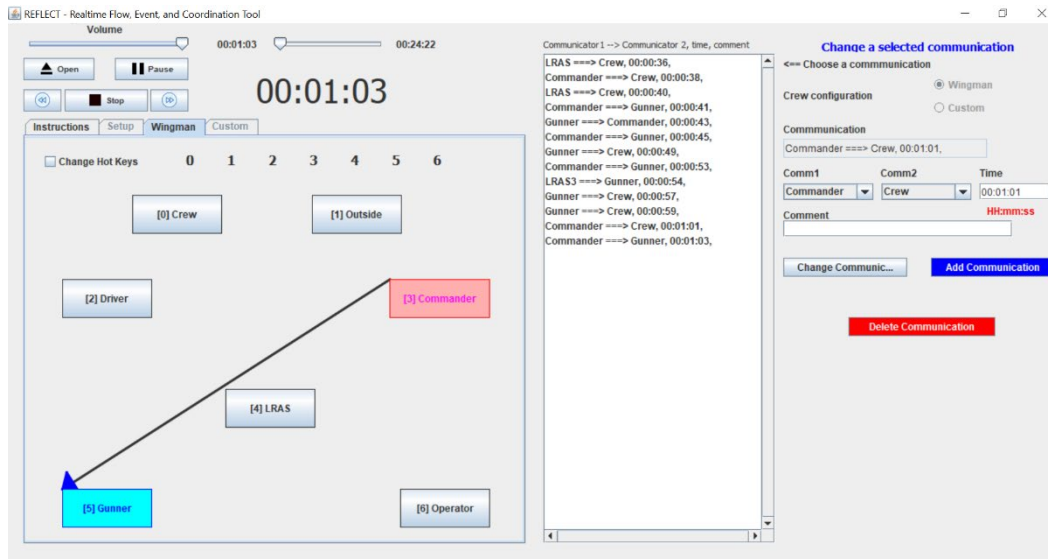


Fig. 5 Example run using REFLECT in support of a Wingman exercise. The left pane shows an example of a sender–receiver message. The center pane displays the sender, receiver, and a timestamp for each communication event. The right pane allows the user to adapt information collected in the center pane.

While one input method for indicating senders and receivers involves clicking the appropriate roles with a mouse, users can also use a keyboard number pad for increased speed and accuracy. For example, in Fig. 4 the input pane reads “[3] Commander” and “[5] Gunner”. This means that the user can simply press 3 and 5 in sequence to create an event with the commander speaking to the gunner. By extension, to create an event where the gunner responds to the commander, one would press 5 then 3.

After an event is entered into the event pane it can then be adjusted by the user to correct any inaccurate information or to add additional information. This is done using both the event pane and the editing pane (in the middle and right side of REFLECT, respectively). First, the user clicks an event in the event pane that should be changed. Then the user can click on the boxes provided in the editing pane to change the sender, receiver, or timestamp for that event. Using the Comment box, the user can also append a comment to the communication event. The user presses the Change Communication button to finalize the changes to the event. If necessary, the user can also add a communication event at the current position in the event stream using the Add Communication button or delete a selected communication event using the Delete Communication button.

In addition, the user can append comments, notes, or even qualitative codes to communication events using the pane on the right side of the interface (Fig. 6). In this way, information can be added for many different purposes. For example, text

can be appended to recorded audio data as a means to skip the step of transcribing speech to text, appending text directly onto communication events and saving time. Alternatively, notes can be added that describe equipment errors or observational notes.

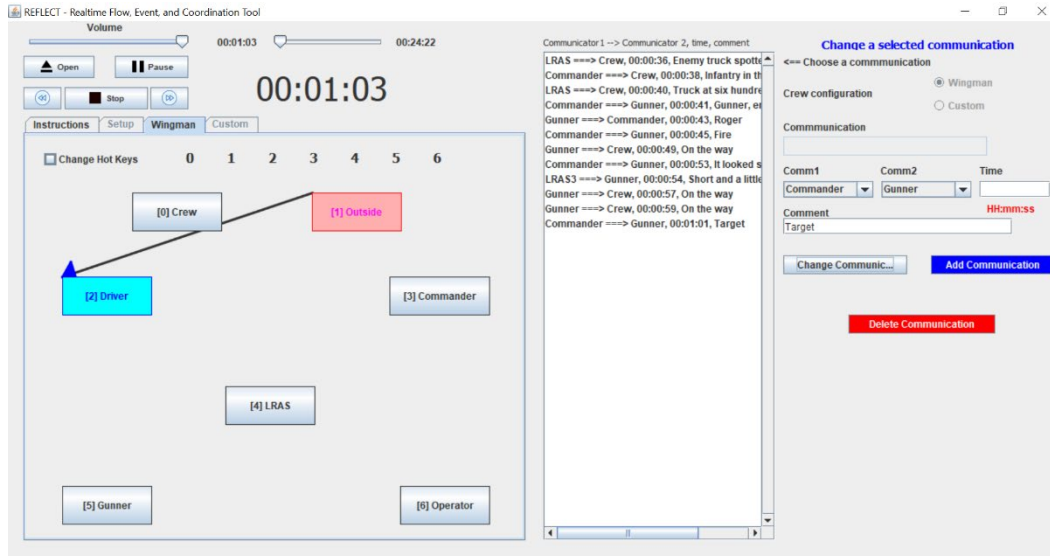


Fig. 6 Communication events that have been transcribed using the right pane. Note each event in the middle pane now has text appended after the timestamp.

1.2.3 Data Output

Once the user finishes collecting data and appending any comments, notes, or codes, the user can close the application by clicking on the X in the upper-right-hand corner of the window. Upon closing the window, REFLECT automatically exports a comma-separated value (.csv) file that can be opened and edited using standard spreadsheet software such as Microsoft Excel (Fig. 7). This exported file contains a list of the senders, receivers, timestamps, and appended notes for all communication events. At this time, closing the REFLECT window is the only way to export data, but an export feature will be added to the interface to circumvent this requirement.

	A	B	C	D
1	Sender	Receiver	Time	Message
2	Commander	LRAS3	7:48	Right there, see him LRAS?
3	LRAS3	Commander	7:49	Yes
4	Commander	LRAS3	7:50	Alright
5	LRAS3	Crew	7:54	Lasing
6	Commander	Gunner	7:56	Gunner, you already got him?
7	Gunner	Commander	7:58	Yeah I already got him
8	Commander	Gunner	7:59	Alright, go ahead and center in on him
9	LRAS3	Crew	8:03	Range is 760 meters
10	Commander	Gunner	8:18	And gunner you're clear to engage
11	Gunner	Commander	8:22	Sorry I'm trying to zoom
12	Commander	Gunner	8:29	Zoom back out
13	LRAS3	Gunner	8:30	Zoom back out, just do two zooms
14	Commander	Gunner	8:37	There you go

Fig. 7 Example data file where data were exported from REFLECT as a .csv file and opened in Microsoft Excel. Column A indicates the sender. Column B indicates the receiver. Column C contains timestamps down to the second, though a future version will account for milliseconds as well. Column D contains any text information that was appended to each communication event.

The output produced by REFLECT can be parsed to identify the number of communication events experienced by each possible sender–receiver pair. Currently, this is done via Excel, but a future version of REFLECT is planned to produce the count of sender–receiver pairs automatically. This can then be organized to show which sender–receiver pairs account for greater proportions of the team’s communication as well as the overall flow of information between team members. An example from the Wingman dataset is provided in Fig. 8, which represents how the type of communication data collected via REFLECT can be used to represent communication patterns within a team.

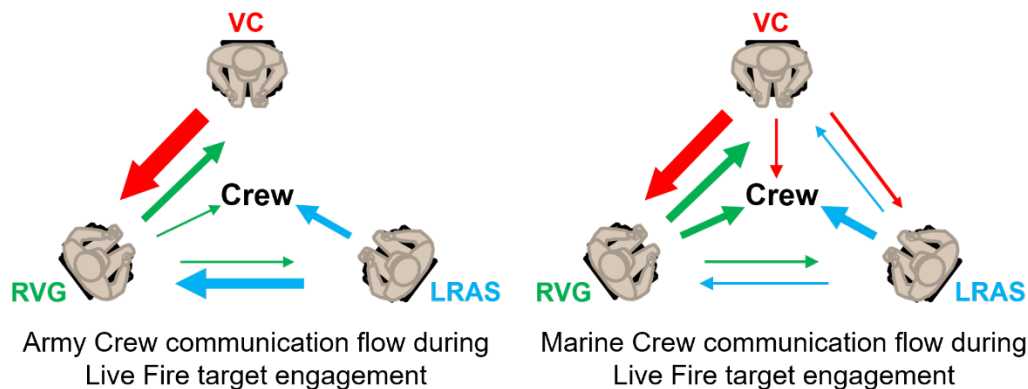


Fig. 8 Communication flow diagrams for Army and Marine crews during human–autonomy gunnery events during target engagement tasks. Thicker lines indicate more communication from that speaker to the indicated recipient.

This approach to mapping communication flow was based on the method of Fischer et al. (2007) but diverged from that method by identifying, rather than ignoring, communication that was not targeted at a specific team member. Some examples of these crew-targeted communications from the live target-engagement corpus were “Enemy scouts at 11 o’clock”, “I’m going for the vehicle”, and “Troops in the open”. By accounting for this nontargeted communication to the crew, it becomes possible to capture information that teams may share to improve situation awareness and update shared mental models (Cannon-Bowers et al. 1993; Wildman et al. 2014).

The data visually represent the flow of communication within the team and reveal some trends. For example, Fig. 8 demonstrates that the Army crew had a rigid communication structure dominated by the vehicle commander (VC) and long-range scout surveillance system (LRAS3 operator) feeding information to the robotic vehicle gunner (RVG), whereas the Marine crew had a more flexible communication structure involving a more balanced distribution of communication within the team. We presented these findings in Schaefer et al. (2019a).

This type of data analysis and visualization is not unique to REFLECT; it can be produced by manually transcribing audio and counting instances of sender–receiver pairs, and it can also be produced (and even expanded upon) using network analysis methods. However, there has not been a method for producing this data in real time without additional hardware, data collection devices, or cumbersome software. Therefore, REFLECT fills a unique role in the practitioner’s toolkit, in that it allows a user to collect communication flow data rapidly and easily.

2. Methods: Usability Assessment of REFLECT

We conducted a usability assessment of the initial version of REFLECT to gather user feedback on the state of the tool to inform subsequent design and development. In this usability assessment, participants were tasked with listening to a selected audio file from the Wingman Joint Capabilities Technology Demonstration (JCTD) November 2018 field exercise in which a crew of Marines from the 1st Marine Logistics Group Combat Skills Training School engaged in a mounted machine-gun live-fire exercise involving a manned–unmanned team.

2.1 Participants

Five participants were recruited from the CCDC Army Research Laboratory workforce. This number is in line with standard usability assessments (Nielsen and Landauer 1993; Nielsen 1994; Baxter et al. 2015).

2.2 Crew Audio Recording

The Wingman JCTD program has served as a use case for developing HAT concepts in live and simulated environments (Brewer et al. 2018, 2019). The November 2018 gunnery event at Fort Benning, Georgia, served as the backdrop for the materials that participants evaluated using REFLECT during this usability assessment. During the event, five-person crews used the Wingman manned–unmanned teaming platform to engage in mounted gunnery operations in accordance with Army gunnery qualification standards (Headquarters, Department of the Army 2015). Additional information about this event and the content of the audio recording, such as information about the crew and the tasks that were performed, is available in Schaefer et al. (2019a; 2019b).

2.3 Training

Training consisted of four portions: introduction to REFFLECT, context of audio recordings, guided voice recognition, and practice session. The goal of context training was to familiarize participants with the voices of the different crew members in the audio recording so that data regarding REFLECT use would indicate participants’ ability to use the system rather than their ability to discern different voices.

In the first portion, participants were introduced to REFLECT. The researcher explained the user interface and features of the tool, such as play and pause, fast-forward and rewind, and marking senders and receivers for communication events. The researcher also briefly demonstrated the features.

In the second portion, participants were introduced to the context of the audio recordings they would be evaluating. To ensure that participants would understand the communication they were listening to, they were provided information about the gunnery task carried out by the Marines. This information involved the Marines’ goals during the task, their roles in the team, and the operational context of their task. This background information helped the user to understand the communication within the Marines team.

In the third portion, the researcher guided participants through an audio recording of the Marines completing a dry-fire practice gunnery exercise. The focus of this training was to increase the participant’s familiarity with each Marine’s voice. Whenever a crewmember spoke, the researcher indicated their role to the participant. Generally, the only Marines speaking during the gunnery engagements were the VC, RVG (responsible for aiming and firing the unmanned vehicle’s weapon), and the LRAS3 operator (responsible for lasing targets and reporting

distances). After the participants indicated a basic familiarity with distinguishing voices of the VC, RVG, and LRAS3 operator, the final portion of training was conducted.

In the final portion, the participants were tasked with using REFLECT to log the sender and receiver for each communication event in an audio recording of the Marines completing a dry-fire practice gunnery exercise. The focus of this portion was on building the participant's familiarity with the process of interacting with REFLECT and logging communication events. Each participant used REFLECT to log practice engagements until they reported feeling comfortable with the interface and with the task.

2.4 Usability Task

For the usability portion of this study, each participant was tasked with using REFLECT to log the sender and receiver pairs for each communication event in an audio recording of the same Marines engaging in a live-fire HAT mounted machine-gun Table VI exercise. The structure of the Marines' task in this live-fire audio file was similar in structure to the dry-fire audio file that participants had been exposed to during practice, ensuring participants were familiar with the context of the task.

2.5 Usability Questions

To gain a better understanding of how specific aspects of REFLECT contribute to the user's experience when listening to and coding crew communication, specific questions were created (Fig. 9). Each question relates to the ease of use of the different features of REFLECT.

While using REFLECT, how easy was it to...

	Very difficult						Very easy
Correctly identify who was speaking?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Accurately mark senders and receivers using the interface?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Keep up with the rate of communication of the team?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Make changes to logged events?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Use REFLECT, overall?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fig. 9 REFLECT usability questions

2.6 Data Preparation

To evaluate the ability of participants to accurately capture communication events using REFLECT, a reference dataset was produced from the same audio recording to formulate a ground-truth accuracy measure. Participant datasets were then compared with this reference dataset. To produce the reference dataset, the sender and receiver for each communication event was annotated by the lead author, who was extensively familiar with the dataset, having attended the gunnery tasks in person. The datasets included all spoken phrases, and excluded nonfluencies and non-information-containing vocalizations. For communication events without a specific receiver, the receiver was indicated as Crew. A timestamp for each communication event was also included in the datasets.

3. Results

The data produced by this study comprised two general categories. In the first, performance data provided insights into how effectively participants used REFLECT during the event-logging task by comparing their REFLECT outputs to the reference dataset described in Section 2.6. The second category involves the usability insights garnered from participants to drive further development of REFLECT. The following sections detail our findings.

3.1 REFLECT Performance

Each participant’s REFLECT output file was compared with the reference dataset for the last two (out of the seven) engagements in the audio file. The last two engagements were selected for analysis because they occurred at the end of the trials, when participants had the most practice with REFLECT and the most familiarity with discriminating between the voices of the gunnery crew in the audio recordings. One participant did not complete this portion due to time constraints and therefore was excluded from analysis.

To calculate performance, hit rates were produced for participants to indicate how often they correctly identified senders and receivers using the reference dataset as a baseline. To calculate these hit rates, participants’ REFLECT output logs were compared with the reference dataset on a timeline that was set to 1-s increments. This allowed events to be compared despite timing differences between them. For example, if the reference dataset contained a message from the VC to the RVG at 1:12, but if the participant’s dataset showed that message at 1:15, it would still be marked as correct under the assumption that the participant experienced a slight delay in adding the event using the interface. However, if the message at 1:15 was indicated as the LRAS3 speaking to the robotic vehicle operator, it would not be correct.[†]

Each participant’s total number of correct senders was calculated by comparing their responses with those of the reference dataset. For example, if Participant A correctly identified 13 senders and the reference dataset contained 17 events, Participant A would achieve a sender hit rate of 76.5%.

Next, receiver hit rates were calculated in the same manner for each event, with one additional consideration. REFLECT allows the user to select Crew as the receiver for nonspecifically targeted messages (e.g., “I see the target”), but review of the data indicates that for many of such nontargeted messages, participants still opted to indicate a crew member as a recipient. In other words, while the receiver they specified could be interpreted as correct during the flow of the recorded conversation, those messages were also not clearly directed at a crew member, so Crew would also have been an acceptable receiver for such events (refer to Fig. 7 for a dialogue example from a crew of Marines during a gunnery field event).

Table 1 indicates two receiver hit rates. The first receiver hit rate indicates the “strict” hit rate, in which participants are penalized for not selecting Crew as the

[†]Note other annotation software packages (e.g., Praat, ELAN, and ANVIL) allow the user to tie timestamps to a given time in an audio file by providing visual waveforms, a seek bar, and other GUI tools that aid in accurate timestamping. Such features are in consideration for addition to future versions of REFLECT.

receiver of a nonspecific message. The second receiver hit rate is provided in parentheses and italics in Table 1. This hit rate, described as “liberal percentage”, indicates the hit rate achieved when, for nonspecific messages, participants specified receivers that could be interpreted to be correct in the context of the crew’s recorded conversation. For example, if the crew’s commander says “Target at 11 o’clock” and the gunner says “I’m locked on target”, it may be strictly correct to state that the gunner’s message is nonspecific since it does not verbally denote a reply to the commander’s message. So, a participant who marks Commander as the recipient for this message might be penalized under the strict definition of hit rate since the gunner’s message is not clearly and specifically directed at the commander. At the same time, the participant could receive credit under the liberal definition of hit rate because the gunner’s message follows right after the commander’s, and thus it could be interpreted as a response to the commander’s information push despite the fact that it was not specifically directed at the commander. Finally, lag times were calculated for all participants. Lag time was calculated as the difference between the start of a communication event and the participant’s REFLECT timestamp measured to the second. In other words, if a communication event began and 2 s later the participant logged an event in REFLECT, the lag time would be 2 s. Average lag times are provided in Table 1 for all participants across each engagement and then in total.

Table 1 Percent effectiveness of identifying sender–receiver pairs using REFLECT across the two test runs for each participant.

Participant	Run	Sender hit rate	Receiver hit rate strict% (liberal%) ^a	Average lag
P2	1	76.5%	35.3% (76.5%)	1.23 s
	2	72.7%	45.4% (72.7%)	2.75 s
P3	1	70.6%	41.2% (70.6%)	1.08 s
	2	72.7%	36.4% (63.6%)	1.63 s
P4	1	58.8%	64.7% (70.6%)	2.1 s
	2	63.6%	54.5% (63.6%)	2.86 s
P5	1	76.5%	52.9% (76.5%)	2.0 s
	2	72.7%	54.5% (72.7%)	2.25 s

^a Strict and liberal scoring are described in the preceding paragraphs.

Note. One participant did not complete this portion due to time constraints and therefore was excluded from analysis.

We note several trends from the table. First, most participants achieved around 70% accuracy with REFLECT. This is notable given that by the time they began coding these engagements, participants had only gained approximately 30 min of experience with the tool and with the voices of the crew members in the audio recording. Second, the liberal receiver hit rates also hovered around 70%,

suggesting that it may take an average user less than an hour to achieve approximately this level of performance. Third, the average lag time for participants was about 2 s ($M = 1.99$ s, $SD = 0.60$ s). Lag times were compounded by two factors: 1) the user may not immediately be aware of the sender or recipient of a communication event, and may be waiting to hear more information to create the event in REFLECT, and 2) participants had different response times to events and some recognized speakers more quickly than others. Again, it is promising that these participants achieved approximately 70% accuracy in identifying sender–receiver pairs with relatively little training and familiarity with the source audio material, though we seek to improve this number with additional improvements to the interface and training.

3.2 Usability Feedback

Usability feedback from participants involved requests for features and potential errors in the tool. Users identified the following critical areas for improvement:

- **Add and reassign hotkeys for REFLECT roles.** By allowing the user to change the input keys used to log information, users would be able to customize the interface in a way that is most sensible to them, likely improving the speed and fluidity of data capture.
- **Customize the position of team member roles on the user interface.** This would make it easier for the user to visualize crew members. For example, if using REFLECT on members of a Wingman crew, the user might reposition the roles on the user interface to match the vehicle seating positions of the crew members or perhaps move more-frequent conversational partners closer together.
- **Add keyboard shortcuts to fast-forward, rewind, and pause audio features.** Like the first point, this would allow the user to interact with REFLECT more fluidly by removing some of the need to shift hand positions between keyboard and mouse.
- **Add a button to undo the last selected role.** If the user presses the wrong role button when trying to log a sender there is no way to undo this action, and the user has to complete the communication event by logging a receiver and removing a communication event after the fact using the “edit communication event” features.
- **Select specific points in an audio file.** The version of REFLECT used by participants allowed participants to fast-forward or rewind in 5-s increments but did not have a seek function to select specific times.

- **Always see the latest communication event in the event pane.** The version of REFLECT being evaluated does not auto-scroll as new events are added, so participants had to scroll down to see the latest events after the event pane had filled. Making the event pane auto-scroll would keep the newest communication events visible.

3.3 Usability Questions

Participant ratings on the usability questions are represented in Fig. 10. Findings suggest that the most difficult aspect of using REFLECT was keeping up with the rate of communication in the audio recording. The extent to which this was due to their relative unfamiliarity with the crew versus the design of the technology is unclear. It may be possible that with more familiarity with the context of the gunnery task and the voices of the crew members this rating would be higher. The next most difficult aspect of using REFLECT was in correctly identifying the speaker of a communication event. Again, this may have been due to the limited experience of participants in this task. Finally, most participants rated REFLECT as slightly above average in terms of how easy it was to use REFLECT overall. While this is not a largely diagnostic item, it is promising that the participants' responses clustered on the easier end of the scale's midpoint.

REFLECT Usability Perceptions

On average, how easy was it to...

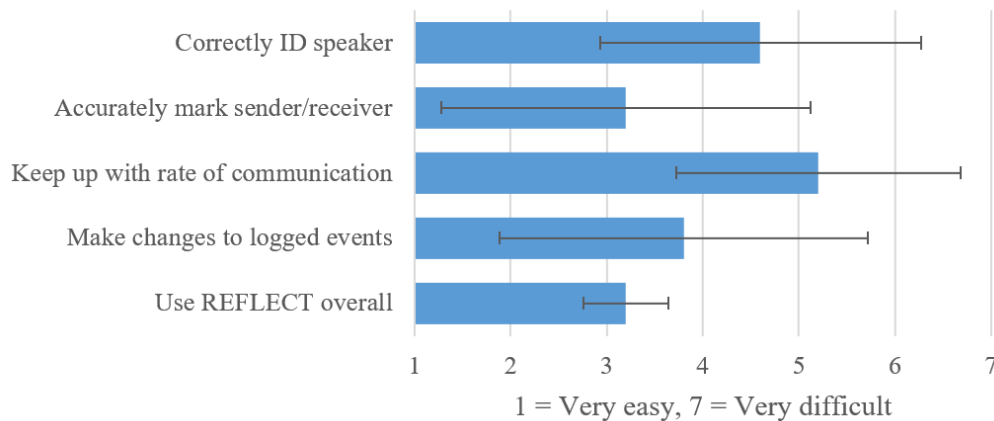


Fig. 10 Usability ratings provide by participants. Error bars indicate 1 standard deviation.

4. Discussion

The performance and usability data collected during the study reveals several implications about the use of REFLECT by human observers. Until REFLECT can be fully automated and integrated with event-logging software, these implications

should be considered during the development of additional features. Primarily, the findings provide insights into how users should be trained to improve their effectiveness with REFLECT and how users should decide what, specifically, constitutes a communication event.

4.1 Training

To accomplish the task and keep up with the team's rate of communication, the REFLECT user needs to develop familiarity with two sets of information. First, the user needs to become familiar with the voices of the team members to positively identify who is speaking at a given time. Second, the user needs to become familiar with the tasks and environment that the team will engage.

4.1.1 Familiarity with Voices

Learning team member's voices allows the REFLECT user to positively identify the source (either sender or receiver) of a communication event. While capturing data, a cognitive burden is placed on the user to make rapid decisions about the source and destination of each communication event, so increased familiarity with individual voices can reduce the cognitive burden and increase the speed at which decisions can be made about communication events. Familiarity also means that the user can make a positive identification when team member speech is altered by changes in tone, inflection, volume, or cadence, among others. These changes can be the result of low or high task demands, so it is important that the user is able to discriminate between voices, especially between those that may sound similar. The data output from REFLECT is only as accurate as the REFLECT user's ability to keep up with the task, so until automatic identification capabilities are integrated with REFLECT, being familiar with each team member's voice is key.

4.1.2 Familiarity with Task Context

It is also important that the REFLECT user becomes familiar with the team's tasks, goals, and overall operational context. When identifying team communication patterns, it is important to be able to correctly identify the sender and receiver of the message. From this data collection, we found that users identified speakers by relying on recognizing the speaker's voice and by gleaning context clues from the communication to infer who was speaking. Because the intended recipient of a communication event is not always obvious (i.e., the speaker may not always identify their intended recipient by name), there may be a higher reliance on context clues to determine the speaker.

4.2 Unitizing Communication Events

When using REFLECT with verbal communication, the user needs to decide how messages will be divided into discrete units (i.e., events). Unitizing communication into discrete events has implications for the quality of the dataset. If the REFLECT user consistently separates communication events based on a certain set of criteria, the dataset can be considered more reliable.

There are generally two approaches to unitizing verbal communication for REFLECT: thought unit (TU) and TT. A TU represents one word, phrase, or utterance that comprises individual thoughts in a discussion (Pavitt 2017). In TU coding, the REFLECT user will separate events into phrases that contain discrete thoughts, or semantic intentions. For example, if one speaker says “Enemy identified, reverse course immediately”, this could be coded as two separate messages since “enemy identified” conveys a different thought (and has a different function) than “reverse course immediately”. The advantage of using TU coding is that it allows for granular insights into what a speaker is doing when speaking, such as asking a question or making a statement. The primary issue with using TU coding stems from distinguishing between events. It can be difficult to consistently separate a conversation into TUs based on how we speak, as it may not be clear where one TU ends and another begins. In these cases, rules and systems can be used to support consistent coding of TUs such as the paradigm devised by Auld and White (1956).

In TT, each time an individual speaks to another, it forms one event, and the next event begins when another speaker joins. TT events are more consistently defined because it is easier to discriminate between units. However, there can be large variations in the amount of information that each unit can contain because a speaker can continue adding information indefinitely (e.g., in the case where another speaker does not take a turn for an extended period).

Participants in the usability study were not instructed to focus on unitizing their communication in any specific way. This was done to observe the approach that they would naturally use. Observation indicated that users tended to use a mix of TU and TT coding. In most cases, participants would start a new communication event whenever another speaker started speaking or when one speaker exhibited a pause between subsequent messages.

5. REFLECT Updated Development

Stemming from participant feedback, several changes have been implemented in REFLECT.

5.1 Change Hot Keys

First, users are now able to change the hotkeys associated with each crew member by using the Change Hot Keys functionality in the input pane (Fig. 11). The user clicks Change Hot Keys and then drags each number to a different role. After clicking the button again, each role has the new number assigned to it. The user can then create a communication event using the new hotkeys. This can be useful for several reasons. For one, crew members may be seated in certain positions, and changing their hotkeys may make it easier to press numbers that correlate spatially to their locations or based on the frequency of communication per role. For another, users may have a preference in the hotkey numbers that allow them to input events most quickly and may wish to have hotkeys that allow them to reach better efficiency. Previously, the button location was not movable. In the latest version of the application, a user can drag the buttons within the window to create a custom layout. When Change Hot Keys is checked, the user can click and hold the right mouse button and drag the crew member button to a new location and release the right mouse button to drop the button into location.

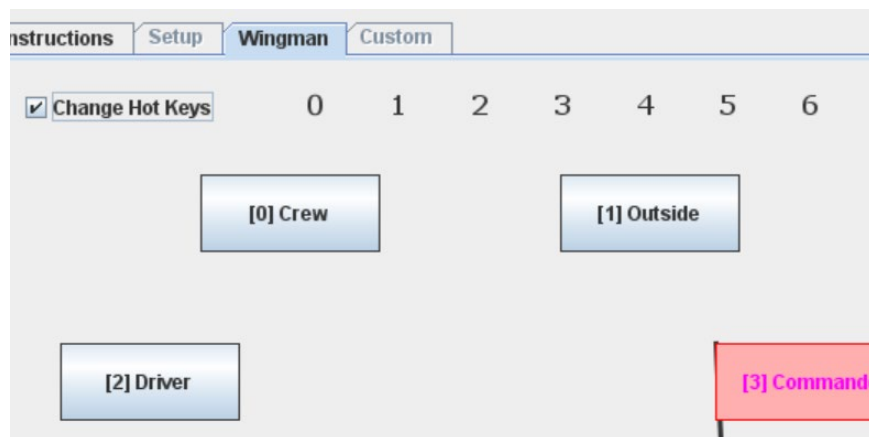


Fig. 11 Zoomed-in screenshot depicts the method to change hotkeys assigned to each role. User clicks Change Hotkeys and then drags each number onto a role.

5.2 Undo Speaker Action

Another feature added to REFLECT is the ability to undo the action of marking a sender and starting a new communication event. Previously, if a user selected the wrong speaker for a communication event there was no way to cancel it, and they

had to finish creating the event by selecting a receiver and removing the communication event after the fact. Now if a user selects the wrong speaker, they can undo this action by pressing the same speaker's key again, which cancels the event.

5.3 Custom Audio Selection

Users are now able to seek to different points in the audio. Previously, REFLECT only allowed users to search through an audio file by clicking Fast Forward or Rewind, which moved the audio file in 5-s increments. Now a slider has been added to the Audio pane, which allows the user to move the slider to any point in the audio file. The latest communication events are now always visible in the event pane. When a user creates enough events to fill the event pane, it will automatically scroll downward as new events are added, ensuring that the user does not have to take their hands off the keyboard to scroll down toward new events.

Features to be added in the near term involve fast-forward and rewind keyboard hotkeys to allow users to search through the audio file without removing their hands from the keyboard. In addition, we plan to allow the user to move the role icons within the input window to match whatever configuration the user desires. REFLECT will also be updated to automatically produce flow maps depicting the communication patterns exhibited during data collection (see Fig. 8). Currently, flow maps must be produced manually using the data exported by REFLECT, but this process can be automated, which will be valuable to users who wish to use REFLECT in training, test and evaluation, and other scenarios that would benefit from having the data available immediately.

5.4 Future Tool Development

Future development of REFLECT will also seek to integrate additional data sources as well as artificial intelligence and machine learning into the tool. If REFLECT can receive input from event loggers that exist in some testbeds, for example, the process of calculating senders and receivers can be automated. This would reduce the potential fallibility of a human observer. However, the accuracy of this automation would also depend on the method used for unitizing communication if such communication is verbal (see Section 4.2). If TT is used, automation is much easier, as push-to-talk button presses can be used as a proxy for tracking conversational turns. However, if TU is used, automation becomes magnitudes more difficult, as the automation must be able to determine when one verbal stream is distinguished by several different TUs. In the near term, TT will be the easiest to

implement, but additional work in the human–computer language domain must be done to make TU distinctions feasible.[‡]

In the near term, it will likely prove easier to integrate a method for automatically detecting the sender of a communication event, such as using push-to-talk key presses to log when each user initiates messages or using microphone input to detect when a team member begins to speak. However, automatic detection of the receiver of a communication event will prove more difficult if the communication medium used by the team is verbal as opposed to digital. In this context a human observer may continue to prove useful, especially if detection of senders can be automated, as this would offset some of the cognitive burden imposed by real-time coding. Following these additions, follow-up testing can be conducted to evaluate whether the changes were effective.

Four other types of features are planned for REFLECT. First, the current version serves as a prototype that allows us to test REFLECT’s use on audio recordings, so the upcoming version of REFLECT will implement support for use on live scenarios such as by including a live timer. Second, the current version outputs timestamps down to the second, but an update will include timing down to the millisecond. Third, we will implement a button that will allow the user to export the data output with timestamps without having to close REFLECT. Fourth, we will add an “Unknown” destination, to allow the user to log a communication event when the intended receiver of the communication cannot be determined.

REFLECT is intended to fill a niche in the tools available to communication researchers in that it is easy to deploy and allows a user to log communication events on the fly, whether from observing a live scenario or via importing an audio recording. In the future, when automated assessment of senders and/or receivers is integrated and when additional event-logging capabilities are developed, REFLECT will automatically parse team communication data to provide immediate insights into a team’s communication patterns. Because communication is so fundamental to teamwork, understanding the flow of communication within a team lets one understand much more about how a team performs and why.

6. Conclusions

The flow of communication within a team reveals insights into team cognition, coordination, cohesion, and ultimately performance (Fischer et al. 2007; Tiferes et al. 2016). REFLECT is designed to capture communication flow within a team

[‡]We refer the reader to additional efforts to understand, parse, and annotate spoken human–robot dialogue: Marge et al. (2016, 2019) and Bonial et al. (2019).

by logging and timestamping the sender and receiver of each communication event. Thus, REFLECT is focused on capturing data that can be used to describe the flow of communication in HATs, which may have diverse structures, capabilities, and goals. REFLECT offers a unique utility in this domain as a means to capture data that can be used alongside other forms of analysis.

We conducted a usability assessment of the initial version of REFLECT to gather user feedback on the state of the tool to inform subsequent design and development. Participant performance outcomes suggested that most participants achieved about 70% accuracy in identifying the senders and receivers of communication events within only 2 s of each event. Further, this level of performance was reached with less than an hour of training and exposure to the voices of the crew members featured in the audio recordings annotated by participants.

Usability insights identified critical areas for improvement relating to hotkeys/shortcuts, error correction, and navigation of audio files. These features are relatively quick to address with further development, and longer-term development will focus on adding features to support automatic detection of communication events, whether they are verbal or nonverbal, in human teams or HATs. Development is ultimately focused on understanding how communication in HATs relates to performance and outcomes, and a clearer understanding of a team's communication patterns will provide a better window into how well they can achieve their goals. Future work will document advances in the development of REFLECT, as well as applications to HATs in simulation and/or field studies.

7. References

- Auld F, White AM. Rules for dividing interviews into sentences. *Journal of Psychology*. 1956;42:273–281.
- Baker AL, Schaefer KE, Hill SG. Teamwork and communication methods and metrics for human–autonomy teams. Aberdeen Proving Ground (MD): CCDC Army Research Laboratory (US); 2019 Oct. Report No.: ARL-TR-8844.
- Baxter K, Courage C, Caine K. *Understanding your users: a practical guide to user research methods*. Burlington (MA): Morgan Kaufmann; 2015.
- Berlo D. *Process of communication: an introduction to theory and practice*. San Diego (CA): Harcourt School; 1960.
- Boersma, P. Praat, a system for doing phonetics by computer. *Glott International*. 2001;5:341–345.
- Bonial C, Traum D, Henry C, Lukin SM, Marge M, Artstein R, Pollard KA, Fouts A, Baker AL, Voss CR. Dialogue structure annotation guidelines for Army Research Laboratory (ARL) human–robot dialogue corpus. Aberdeen Proving Ground (MD): CCDC Army Research Laboratory (US); 2019 Oct. Report No.: ARL-TR-8833.
- Brewer RW, Cerame EJ, Pursel ER, Zimmermann A, Schaefer KE. Manned-unmanned teaming: US army robotic wingman vehicles. *Proceedings of the International Conference on Applied Human Factors and Ergonomics*; 2018 July 21–25; Orlando, FL. Cham (Switzerland): Springer; c2018.
- Brewer RW, Walker AJ, Pursel ER, Cerame EJ, Baker AL, Schaefer KE. Assessment of manned-unmanned team performance: comprehensive after-action review technology development. *Proceedings of the International Conference on Applied Human Factors and Ergonomics*; 2019 July 24–28; Washington (DC). Cham (Switzerland): Springer; c2019.
- Butts CT. Social network analysis: a methodological introduction. *Asian Journal of Social Psychology*. 2008;11(1):13–41.
- Cannon-Bowers JA, Salas E, Converse S. Shared mental models in expert team decision making. In: Castellan NJ, editor. *Individual and group decision making: current issues*. London (England): Psychology Press; 1993. p. 221–246.
- Cooke NJ, Gorman JC, Myers CW, Duran JL. Interactive team cognition. *Cognitive Science*. 2013;37(2):255–285.

- Fischer U, McDonnell L, Orasanu J. Linguistic correlates of team performance: toward a tool for monitoring team functioning during space missions. *Aviat Space Environ Med.* 2007;78(5):B86–B95.
- Headquarters, Department of the Army. Training and qualification, crew. Washington (DC): Headquarters, Department of the Army; 2015 Mar. Training Circular No.: TC 3-20.31.
- Kipp, M. (2012). Multimedia annotation, querying, and analysis in ANVIL. In: Maybury, MT, editor. *Multimedia information extraction*. Hoboken (NJ): John Wiley & Sons; 2012. p. 351–367.
- Marge M, Bonial C, Pollard KA, Artstein R, Byrne B, Hill SG, Voss C, Traum D. Assessing agreement in human-robot dialogue strategies: a tale of two wizards. In: Traum D, Swartout W, Khooshabeh P, Kopp S, Scherer S, Leuski A, editors. *Proceedings of the 16th International Conference*; 2016 Sep. Cham (Switzerland): Springer; c2016. p. 484–488.
- Marge M, Nogar S, Hayes CJ, Lukin SM, Bloecker J, Holder E, Voss C. A research platform for multi-robot dialogue with humans. 2019. arXiv:191005624. arXiv preprint.
- Marlow SL, Lacerenza CN, Paoletti J, Burke CS, Salas E. Does team communication represent a one-size-fits-all approach? A meta-analysis of team communication and performance. *Organizational Behavior and Human Decision Processes.* 2018;144:145–170.
- Newman MEJ. *Networks: an introduction*. Oxford (UK): Oxford University Press; 2010.
- Nielsen J. Heuristic evaluation. In: Nielsen J, Mack RL, editors. *Usability inspection methods*. Hoboken (NJ): John Wiley & Sons; 1994.
- Nielsen J, Landauer TK. A mathematical model of the finding of usability problems. *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems*; 1993. <https://doi.org/10.1145/169059.169166>.
- Pavitt C. Reliability, unitizing. In: Allen M, editor. *The SAGE encyclopedia of communication research methods*. Thousand Oaks (CA): SAGE Publications, Inc; 2017. p. 1422–1424.
- Sacks H, Schegloff EA, Jefferson G. A simplest systematics for the organization of turn-taking for conversation. *Language.* 1974;50(4):696–735.

- Salas E, Wilson KA, Murphy CE, King H, Salisbury M. Communicating, coordinating, and cooperating when lives depend on it: tips for teamwork. *The Joint Commission Journal on Quality and Patient Safety*. 2008;34(6):333–341.
- Schaefer KE, Baker AL, Brewer RW, Patton DJ, Canady JD, Metcalfe JS. Assessing multi-agent human–autonomy teams: US Army robotic wingman gunnery operations. In: George T, Islam MS, editor. *Proceedings of SPIE Defense + Commercial Sensing*; 2019a Apr 14–18; Baltimore, MD. p. 109822B. doi: 10.1117/12.2519302.
- Schaefer KE, Brewer RW, Baker AL, Pursel ER, Gipson B, Ratka S, Giacchi J, Cerame E, Pirozzo K. US Army Wingman Joint Capability Technology Demonstration (JCTD): initial Soldier and Marine feedback on manned–unmanned gunnery operations. Aberdeen Proving Ground (MD): CCDC Army Research Laboratory (US); 2019b Mar. Report No.: ARL-TR-8663.
- Shannon CE. A mathematical theory of communication. *Bell Syst Tech J*. 1948;27(3):379–423.
- Tiferes J, Hussein AA, Bisantz A, Kozlowski JD, Sharif MA, Winder NM, Allers AN, Cavuoto L, Guru KA. The loud surgeon behind the console: understanding team activities during robot-assisted surgery. *J Surg Educ*. 2016;73(3):504–512.
- Wildman JL, Salas E, Scott CPR. Measuring cognition in teams: a cross-domain review. *Human Factors*. 2014;56(5):911–941.
- Wittenburg P, Brugman H, Russel A, Klassmann A, Sloetjes H. ELAN: a professional framework for multimodality research. *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*; 2006 May; Genoa, Italy. p. 1556–1559.
- Wren J. WindowBuilder. Ottawa (Canada): Eclipse; n.d. [accessed 2020 Feb 3]. <https://www.eclipse.org/windowbuilder/>.

List of Symbols, Abbreviations, and Acronyms

ARL	Army Research Laboratory
CCDC	US Army Combat Capabilities Development Command
.csv	comma-separated value
GUI	graphical user interface
HAT	human–autonomy teaming/team
JCTD	Joint Capabilities Technology Demonstration
LRAS3	long-range scout surveillance system
REFLECT	Real-time Event, Flow, and Coordination Tool
RVG	robotic vehicle gunner
SA	situation awareness
TT	turn-taking
TU	thought unit
VC	vehicle commander

1 DEFENSE TECHNICAL
(PDF) INFORMATION CTR
DTIC OCA

1 CCDC ARL
(PDF) FCDD RLD CL
TECH LIB

1 CCDC ARL
(PDF) FCDD RLH B
T DAVIS
BLDG 5400 RM C242
REDSTONE ARSENAL AL
35898-7290

1 CCDC ARL
(PDF) FCDD HSI
J THOMAS
6662 GUNNER CIRCLE
ABERDEEN PROVING
GROUND MD
21005-5201

1 USAF 711 HPW
(PDF) 711 HPW/RH K GEISS
2698 G ST BLDG 190
WRIGHT PATTERSON AFB OH
45433-7604

1 USN ONR
(PDF) ONR CODE 341 J TANGNEY
875 N RANDOLPH STREET
BLDG 87
ARLINGTON VA 22203-1986

1 USA NSRDEC
(PDF) RDNS D D TAMILIO
10 GENERAL GREENE AVE
NATICK MA 01760-2642

1 OSD OUSD ATL
(PDF) HPT&B B PETRO
4800 MARK CENTER DRIVE
SUITE 17E08
ALEXANDRIA VA 22350

ABERDEEN PROVING GROUND

13 CCDC ARL
(PDF) FCDD RLH
J LANE
P FRANASZCZUK
K MCDOWELL
K OIE
K SCHAEFER-LAY
A MARATHE
FCDD RLH F
J GASTON
FCDD RLH FA
A DECOSTANZA
FCDD RLH FD
A BAKER
FCDD RLH FE
D HEADLEY
FCDD RLH P
A EVANS
FCDD RLS ED
Y CHEN
FCDD RLV A
R BREWER