

Autonomy Voice Assistant for NPAS (NASA Platform for Autonomous Systems)

Lucian Murdock, Fernando Figueroa, PhD., and Lauren Underwood, PhD.

NASA Stennis Space Center

Abstract

A prototype voice interaction system, Autonomy Voice Assistant (AVA), is described in this paper. AVA is designed to seamlessly integrate into the NASA Platform for Autonomous Systems (NPAS), an autonomy software platform, and to enable an operator to interact with NPAS autonomy applications through voice conversations. By integrating AVA with NPAS, a major enhancement to NPAS applications is facilitated, enabling interaction through natural language expressions. An AVA prototype has been designed incorporating two principles: (1) self-containment (no external data or computations required), and (2) a readily modifiable, reconfigurable, and flexible architecture. By using voice messages in an NPAS application, an additional layer of user interface capability is enabled, thereby enhancing a user's overall experience. Advancements, over the past several decades in speech recognition and natural language processing technologies has made it possible for AVA to implement robust messaging capabilities while still being lightweight. The main objective of incorporating a voice assistant like AVA is to augment the number and effectiveness of interactions a user has with a system that typically uses mouse-based interaction, while simultaneously enriching the user experience and providing heightened system awareness.

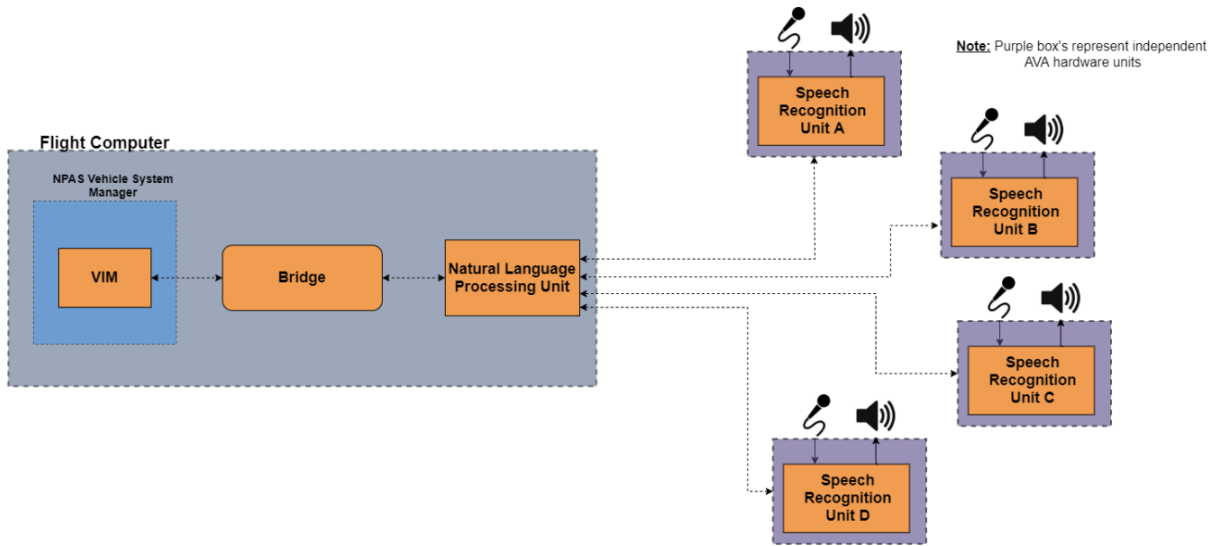


Figure 1. AVA Multi-Client Architecture

Introduction

In the world of computational intelligence, voice activated assistants are becoming widely popular for many systems due to the value added provided by enhancing human-computer interactions. At NASA's Stennis Space Center (SSC) a preliminary prototype of an Autonomy Voice Assistant

named “AVA” has been developed that can be integrated into the NASA Platform for Autonomous Systems (NPAS). NPAS enables development of autonomous “thinking” applications that can be rapidly deployed [1-3]. NPAS has been developed using the Gensym’s G2 environment, providing real time expert/knowledge system capabilities [4]. NPAS is currently being implemented for an autonomous operation application at NASA SSC’s High-Pressure Gas Facility, which includes the ability to analyze, diagnose, and prevent potential failures as they arise.

The Autonomous Systems Laboratory (ASL) at SSC is now working to implement NPAS autonomous operations for “Gateway” (NASA’s future station to orbit the Moon). Leveraging NPAS provides Gateway with comprehensive health and operational awareness as well as enables the ability to operate with any desirable degree of autonomy [1]. Autonomy is required for crew and mission safety in deep space. AVA is a real time assistant designed to make interactions between any NPAS application and users efficient and intuitive. AVA enables operations by voice interactions, which allows the user (e.g. crew of a spacecraft) to spend less time getting to a computer terminal and filtering through a user interface, to achieve a desired activity.

Speech interaction capabilities requires the ability to: (1) translate speech to text, (2) understand the meaning of the text, (3) use that understanding to interact with the G2/NPAS application, which can, (4) respond to a query or confirm an action via voice. An important requirement in the design of AVA was that this all had to be done offline without using any of the cloud-based assistants that are typically used. Figure 1 shows the AVA Multi-Client Architecture.

A key strength of the AVA implementation is the Voice Interaction Module (VIM) within NPAS (Figure 1). The VIM enables effortless interaction with any NPAS application (e.g. a habitat module), due to the comprehensive object-oriented model within NPAS. Each NPAS model consists of a variety of objects representing every item of the system; these objects are used within the reasoning and decision making abilities associated with autonomy. Each object has a name (e.g. Battery-xy) and named attributes (Temperature), so one can use AVA to address “the temperature of battery-xy” as a natural language expression instead of having to manually interact with the NPAS objects through the use of a computer terminal. Language dictionaries can be automatically created by the NPAS application itself (not yet implemented at this time), and updated automatically in real time as the NPAS application evolves. Furthermore, a natural extension is to develop VIM units specialized in certain domains of the NPAS application. For example, there could be an AVA PROPULSION, or AVA Environmental Control and Life Support System (ECLSS). This would help compartmentalize voice interactions to suitable experts and degrees of information fidelity.

This paper describes details of how each individual component of AVA has been developed, as well as an evaluation of the operation and performance of an initial proof-of-concept prototype that allows interaction with an NPAS application.

System Architecture

The focus of the AVA architectural design was to enable the following: (1) user alerts (e.g. crew of a space habitat) of any issues, (2) answers to queries pertaining to state of any subsystems or components, and (3) real time voice interactions with different systems. The AVA architecture

can be broken down into 4 main elements (see Figure 1). First, the Speech Recognition Unit (SRU), which allows AVA to translate continuous microphone input into text. Second, the Natural Language Processing Unit (NLP), which provides the capability to understand the intent of messages that come from the SRU using its trained models. Third, the Bridge which acts as the “middle man”, enabling the external AVA components to interact with the G2 Environment; and finally, the Voice Interaction Module (VIM), located inside NPAS, which completes the user’s desired action, and then sends back a message that is synthesized into audio.

The goal of AVA is to readily enable interactions between a human operator and any NPAS/G2 application, while simultaneously maximizing re-use and configuration options. In the case of Figure 1, the AVA architecture is denoted as a NPAS Vehicle System Manager (VSM) for Gateway. The proposed architecture represents a flexible and interchangeable design, applicable for integrating any system requirements. Transmission Control Protocol (TCP) ports, enabling local network communication, are used to allow each individual component to communicate with one another; in this way, each component could potentially be placed on a separate computer and still communicate (provided there was a local network). For example, the NPAS application (performing autonomous operations) would contain the VIM. The VIM would be a module inside of NPAS and includes the Bridge located as a separate program, all housed inside the flight computer. The embedded AVA hardware architecture, like today’s smart assistants, could be placed wherever needed (i.e. any component) with the ability for the NLP and SRU to communicate with NPAS by sending messages through the Bridge. The messaging protocol between the individual components enables a flexible multi-client architecture (Figure 1).

Types of Messages

Since the VIM integrates directly into G2 and the NPAS environment, the AVA system could support a wide variety of messages and could correspondingly fit a variety of mission profiles. For example, any processes that would require manually sifting through the NPAS software to find certain information, could, using AVA, be automatically accomplished. A small subset of possible messages and use cases are noted in Figure 2.

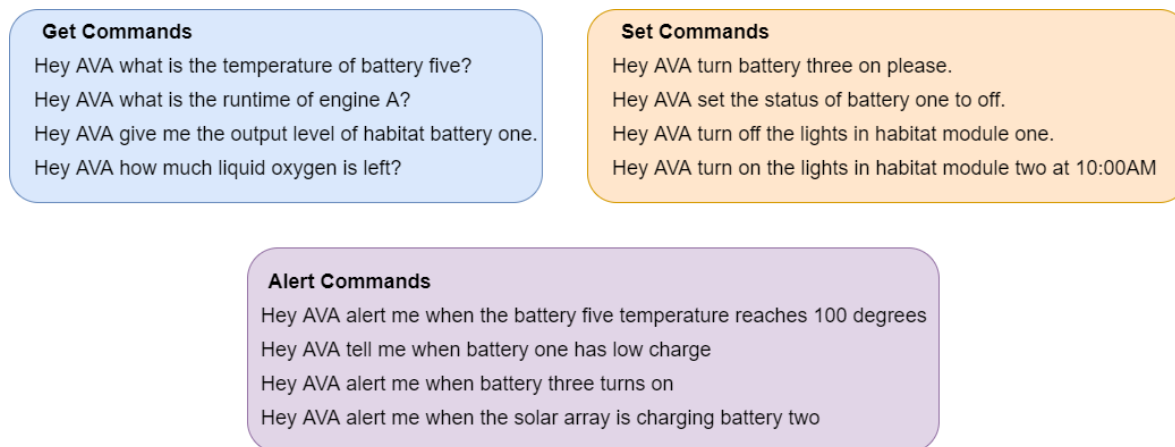


Figure 2 Possible Interactive Messages

In Figure 2, three major conversation types are seen: get, set, and alert. The Gensym G2 environment is object oriented, and components, like actual batteries on the Gateway, would be represented as objects in G2/NPAS. This enables the ability to create generic messages structures in AVA that could then be used to get the value or attribute of any object currently in the G2 environment. The AVA prototype discussed in this paper specifically focuses on the “Get” and “Set” messages, but AVA would not be limited to these two types of messages. “Alerts” could also be used to keep a user updated with status information of the NPAS application (e.g. critical time to depletion of power), audio alerts can satisfy user needs to be made aware of important information. The NLP module can process a wide variety of message types, and additional message types could be created depending on the needs of the NPAS implementation.

Overall Interaction Flow

With a proper communication protocol in place, the overall flow of the AVA system is very linear and easy to understand. The flowchart shown in Figure 3 walks through the major actions each module executes and how communication among the modules occur.

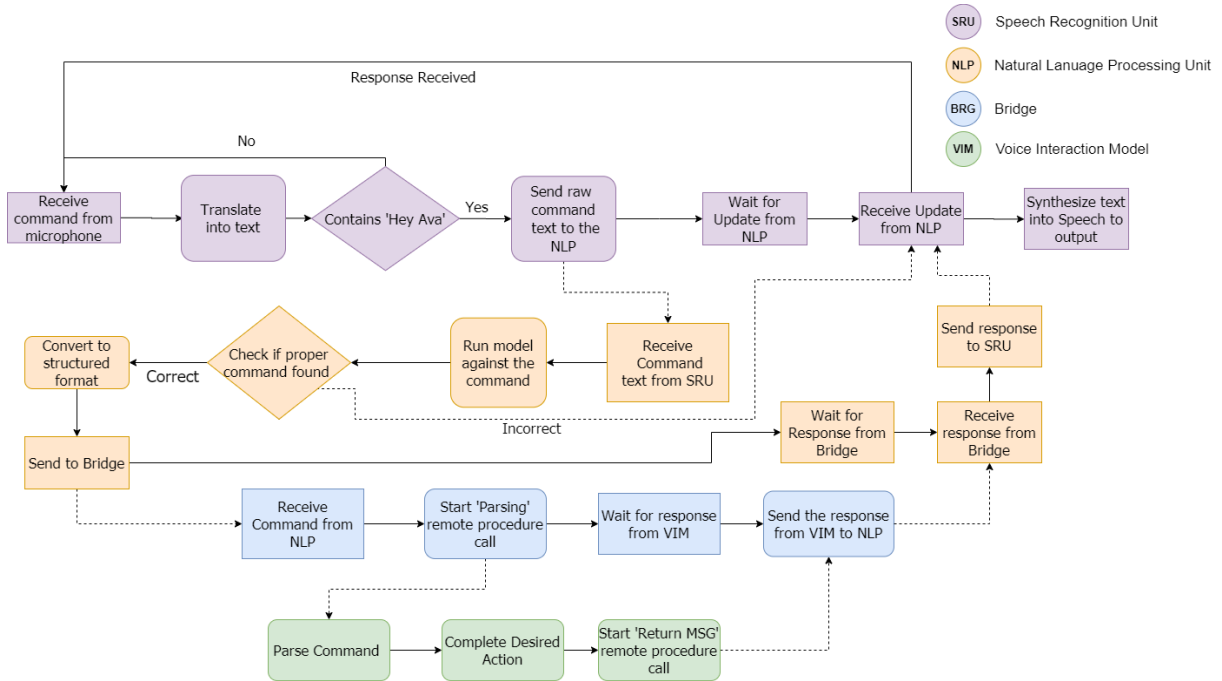


Figure 3 AVA Interaction Loop

The AVA system loop (Figure 3) begins at the Speech Recognition Unit. The SRU listens to the input stream of the microphone and translates incoming speech into a stream of text. If a message containing “Hey AVA” is inside this text, then the raw text is sent over to the NLP. Inside the NLP, the trained Rasa model is used to parse the text and determine the intent based on the user’s speech [5]. That intent might be a simple query, such as “what is the status of a certain battery” or a message to turn off or on a specific system. The NLP’s goal is to determine what the proper intent is, and then convert the message into a standardized format (called a message directive) that the VIM can decipher inside the G2 environment. The AVA message directive is then sent to the

Bridge, and then the Bridge reads in the message and calls the parsing remote procedure call (RPC) function located inside the VIM. The VIM processes the message (e.g. deciphers the intent) and calls the appropriate return RPC function, so the response is sent back to the NLP. Finally, the NLP sends the response to the SRU which synthesizes audio output through the system speakers. The SRU can then accept new microphone input and start the AVA processing loop over again.

This AVA system enables a hands-off approach to interact with the NPAS application, and avoids issues associated with stopping whatever an operator is doing to do so. It is important to note that the entire AVA process works completely independent of any external networks or cloud libraries. Everything AVA needs to operate can all function offline and communicate through a local network.

Conclusion

Over a period of a several of months, a successful prototype of an Autonomy Voice Assistant has been developed. Using speech translation and natural language processing, a user can now use natural language expressions to interact with a NPAS/G2 environment. The AVA prototype that has been developed operates without the need for any external computational capabilities (cloud resources) and uses a modular and flexible architecture. This prototype embodies the foundation for further improvements and enhancements, with an anticipated future goal of creating a production level AVA system. The prototype modularity provides the capability to replace and improve any single AVA component without having to redesign the entire system. Integrating an AVA system into NPAS will add another layer of autonomy and safety for a wide range of potential NPAS applications, i.e. Gateway. In addition to enabling hands-free interaction with an NPAS environment through voiced messages, and capabilities such as audio alerts to the user, AVA could prove valuable in assisting NPAS's goal of providing a safe and reliable software platform for complex deep-space exploration.

References

- [1] Fernando Figueroa, Lauren Underwood, Mark Walker, and Jonathan Morris, "NASA Platform for Autonomous Systems (NPAS)," AIAA 2019-1963, AIAA SciTech Forum, San Diego, California, January 7-10 2019.
- [2] Fernando Figueroa, Lauren Underwood, Ben Hekman, and Jonathan Morris, "Hierarchical Distributed Autonomy: Implementation Platform and Processes," IEEE 2020 Aerospace Conference, Yellowstone Conference Center, Big Sky, Montana, USA, March 7-14.
- [3] NASA Platform for Autonomous Systems (NPAS) NASA TechPort page
<https://techport.nasa.gov/view/94884>
- [4] <https://ignitetechnology.com/gensym/>
- [5] <https://rasa.com/>