Automation of IoT based Decision Making with Uncertainty

Somiya Metu, Adrienne Raglin and Dawn Lott

March 2, 2020

Automation of IoT based Decision Making with Uncertainty

Somiya Metu[1]*, Adrienne Raglin[1] Dawn A. Lott[2],
[1]Army Research Laboratory, 2800 Powder Mill Rd, Adelphi, MD, USA 20873
[2]Delaware State University, 1200 N. DuPont Highway, Dover, DE, USA 19901

ABSTRACT

There is an increased dependence on IoT devices to support teams of humans and agents in a multi-domain battle. There are numerous challenges in the reliance of data from these IoT devices. One critical challenge is the uncertainty related to the devices and data from devices. The uncertainty may significantly affect decision making that relies on this data. In this paper, we use a set of parameters to represent an uncertainty value associated with simulated IoT devices. We discuss Sentry Agents (SAGE), which is a dynamic multi-agent-based framework for system automation. SAGE is used in the construction of various scenarios for decision making involving IoT devices. These scenarios provide a platform to investigate factors that contribute to uncertainty from IoT devices and data when decision tasks are performed.

Keywords: Decision Making, Uncertainty, Multi-Domain Battle, Agent Based Automation

1. INTRODUCTION

There is an increased reliance on IoT devices in a multi-domain battle that include operations crossing land, air, sea, space, and cyber. Uncertainty can come from different sources which poses a serious challenge for decision making in military applications particularly given these complexities. In this paper, we discuss different sources of uncertainty and discuss SAGE, which is a multi-agent-based framework from the Naval Research laboratory. SAGE Framework is utilized to construct a scenario to represent a model of decision making with uncertainty arising from different factors related to IoTs. The model attempts to simulate how knowledge of uncertainty can affect decision making. The decision maker and the different sources of uncertainty are represented via agents in the SAGE framework. In the scenario, the decision maker makes simple decisions after receiving input from different sources of uncertainties. The scenario presented in this paper is one implementation to demonstrate a task and how it can be affected by uncertainty. From this implementation, others can be generated to investigate how various models of decision making and the factors, such as uncertainty, can have an influence. Decision making and uncertainty are complex concepts; therefore, we utilize one task and one type of uncertainty with predefined variables for this initial stage of the work. In addition, this phase of the work simulates the humans and device behaviors and information, allowing us to explore dependent/independent variables and possible outcomes. Analysis and evaluation of the results from these automations are shown. Supporting data is mentioned to frame the results and the plans for future investigations.

2. IOT AND POTENTIAL SOURCES OF UNCERTAINTY

Over the past few years, IoTs have found strong military applications. It has seamlessly connected devices to help soldier maintain and utilize equipment effectively. Despite this connectivity, there exists uncertainty with the use of IoTs. We have selected four main sources of uncertainty that can be linked with the use of IoT devices.

1. IoT Device: An IoT device has an associated uncertainty with it, there is no guarantee that a device functions properly at all times which can lead to issues of unreliability at times. For example, issues with the device performance from device malfunctions or antiquated devices.
2. Information (from the Device): There is an associated uncertainty with the data coming from the IoT device. There is a possibility of data from the device could be out of date or even corrupted.
3. Network: The IoT devices depend on the network for data transmission. There could be instances where the network may be down or intermittent thus creating a possibility of data omission or corruption.
4. Visualization - Visualization of the data from IoT devices is an important factor for comprehension. A visualization mechanism may not be the appropriate one and may fail to convey the right message.

In order to investigate decision making with uncertainty, we have made use of the SAGE framework. We used SAGE to explore representations of sources of uncertainty for decision makers. The next section details the SAGE framework.

3. SAGE

SAGE[1] is a multi-agent-based framework for system automation from the Naval Research Laboratory. It is language and operating system agnostic. It provides for dynamic construction and deconstruction of agents, hence providing a greater level of system control. The SAGE framework allows for language agnostic extensibility and enhancement to support integration of new systems, tools and capabilities. SAGE provides for storage of network states to facilitate automation of large-scale systems and also speeds up automation by quickly restoring to previously saved network states.

An instance of a SAGE runtime consists of one instance of a SAGE Server and one or more SAGE Node instances running on either the Server computer or on remote computers with network connectivity to the Server Computer. In addition, a SAGE Node instance may have multiple SAGE agents which may each have multiple behaviors. The picture below depicts an instance of the SAGE runtime:

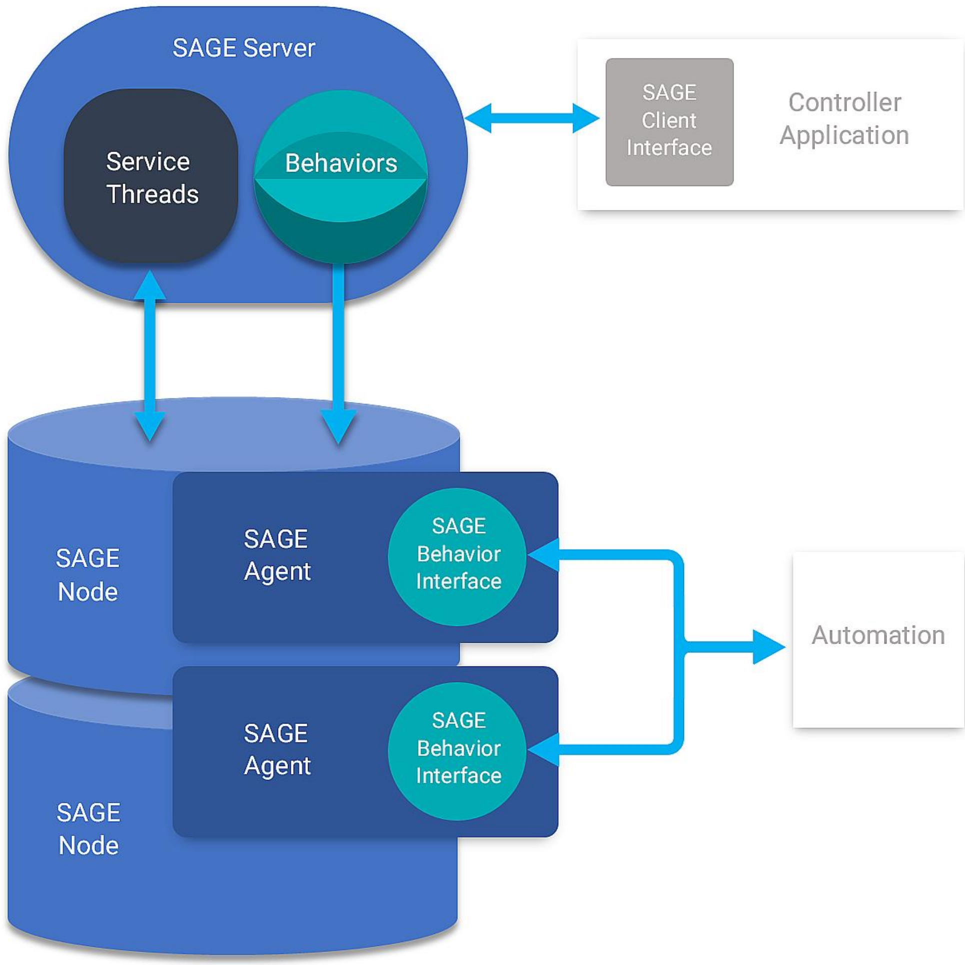

Figure 1. SAGE Runtime instance.

The various SAGE components are described in detail below.

3.1 SAGE Server

SAGE framework contains a console-based SAGE server and a Windows GUI-based SAGE server. The SAGE server is responsible for managing communication amongst the agents in addition to external applications and agents. It provides an interface for external systems to create/destroy and configure agents. It also provides for agents to query other agents in the agent network. One of the important functionalities of the SAGE Server is to provide external applications with information about the agent network.

3.2 SAGE Node

SAGE Node runs on a machine with a unique name. A SAGE Node acts as a container for SAGE Agents. It connects to the SAGE Server to realize requests from agents. It is also responsible for managing the execution of agent's behaviors.

3.3 SAGE Agent

SAGE Agent is contained within a SAGE Node. Each Agent can have a set of beliefs and a set of behaviors. An Agent is cognizant of its belief(s). The "belief" or in other words "state" is represented as name/value pair. Behaviors are Agents capabilities that it can perform. The Agents can have a set of behaviors that are added to the agent after its creation. At any given time, an agent can be active or inactive. A SAGE Agent always maintains its current state. A SAGE Agent has a unique name within the Node it resides in order to be identifiable. SAGE agents can communicate with each other using messages within the SAGE Framework.

3.4 SAGE Controller

The SAGE Controller helps interact with the SAGE Server. The controller can be utilized for a varied number of actions including create, activate, deactivate, configure and message agents.

The next section describes the usage of SAGE components to realize a test scenario.

4. DECISION MAKING WITH UNCERTAINTY UTILIZING SAGE

We utilize SAGE to construct an initial test scenario where agents make a decision based on Uncertainty of Information (UoI) value within the SAGE framework. There are several reasons to choose SAGE as a platform to create automation scenarios. SAGE is a multi-agent system that allows for creation of multiple virtual agents where each agent can act independently and collaborate to perform an action. For our test scenario, we need agents to exists independently and also work in tandem to make a decision based on certain measures of uncertainty. SAGE's framework provides the flexibility to achieve this.

- SAGE allows connection and integration with external systems and tools. This extensibility feature of SAGE helps empower the agents to utilize external tools and software applications within a given scenario. This feature is used to allow agents to connect to IoT application for actuation purposes.
- SAGE provides a way to add different types of behaviors to the agents. This way agents can have varying levels of intelligence and deal with unexpected circumstances. This capability within SAGE helps us build complex scenarios for further testing.

- Last but not the least, SAGE's infrastructure supports introspection where agents are cognizant of other agents and their corresponding beliefs and behaviors. This is a helpful capability in our scenario generation.

A scenario that models decision making by agent based on a given information is detailed below:

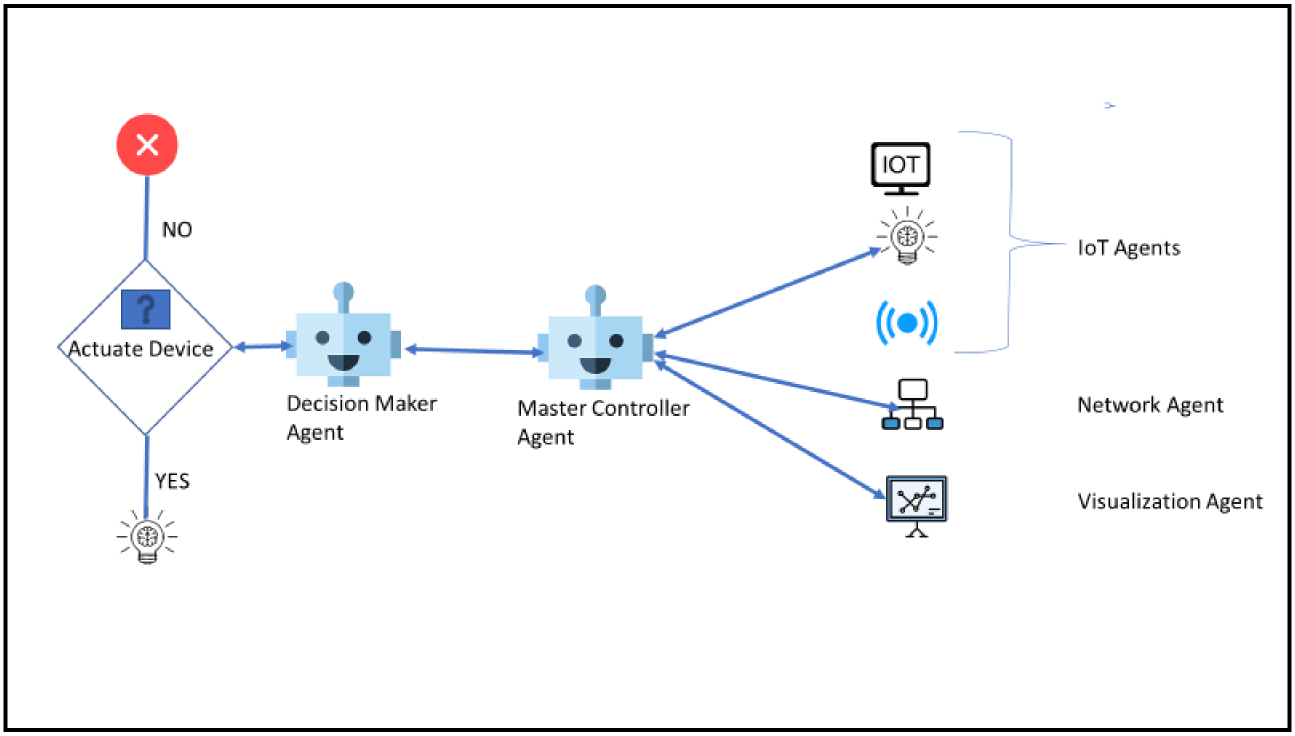

Figure 2: Decision Making Scenario

4.1 Scenario Construction

Our Scenario in the SAGE framework is depicted in Figure 2. It comprises of a Decision-making Agent, a Master Controller Agent, Network Status Agent, Visualization Agent and multiple agents representing IoT devices.

- The Decision-making agent has an action to perform based on an Uncertainty of Information value. In this scenario, performing an action equates to making a decision.
- The Master controller Agent is responsible for communicating with IoT agents, the Visualization agent and the Network Agent to garner their statuses and store the information.
- The IoT agent represents an IoT device, provides the status of the device and its associated device information.
- The Network agent represents the status of the network.
- The Visualization agent represents the quality of the visualization system that presents the information.

The SAGE Server application is used to create a SAGE Node. After the instantiation of a SAGE Node, the agents are created and the necessary behaviors are added to the agents that make them unique with their pre-defined characteristics and attributes. The picture below depicts the graphical user interface of the SAGE Server application. The interface shows all the agents contained within a Node.

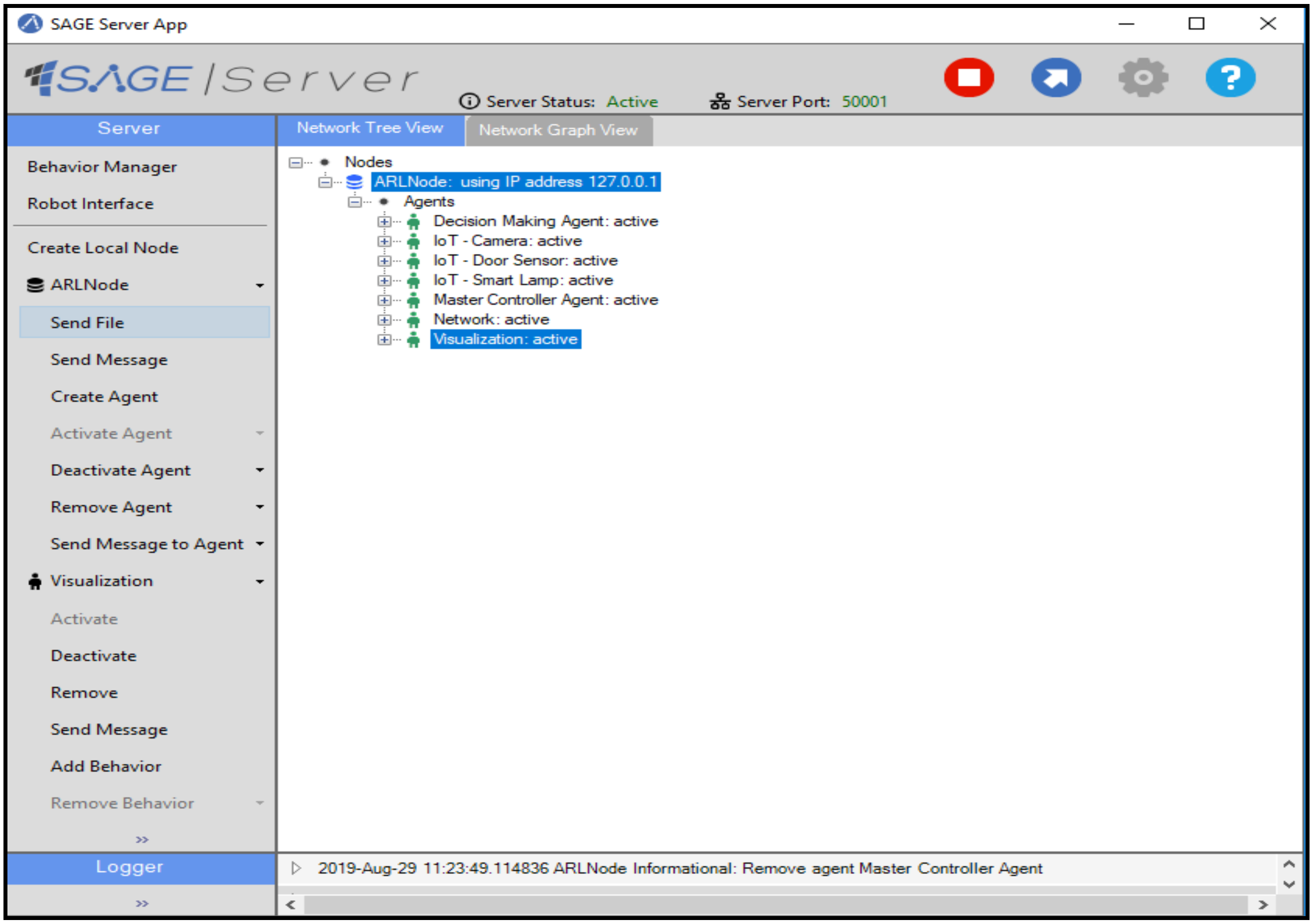

Figure 3: SAGE Server Application

The orchestration of the events in the scenario have been scripted inside a Robot file. The Robot file makes use of the built-in SAGE Keywords; for example, ‘Start SAGE’, ‘Create Agent.’ These keywords support native Robot Framework [1][3] integration as a SAGE controller mechanism for executing commands such as to create an agent and activate an agent. The events in the scenario commences with the Decision Maker Agent querying for information on IoT device(s) for the purpose of actuating a device. The Decision Maker agent reaches out to the Master Controller Agent for information. The Master Controller Agent then polls for information from the Agents representing IoT device(s), the Network agent and the Visualization agent.
Those agents provide the Master Controller Agent with their statuses. This information is compiled by the Master Controller Agent and routed back to the Decision Maker. The Decision Maker Agent

then makes a decision for actuation of an IoT device based on the information provided by the Master Controller Agent.

5. SIMULATION RESULTS

In the scenario described above, the Decision Maker polls the Master Controller Agent to retrieve the statuses of different agents. The Master Controller Agent then communicates back the statuses of agents to the Decision Maker. The Decision Maker receives a combination of uncertainties regarding the statuses. These agents represent IoT devices. IoT Device Data, Visualization of IoT Data and Network for IoT devices. For this simulation we have simplified the uncertainties as levels High(H), Medium(M), or Low(L). In order to classify the combinations, we followed a generic pattern in the following sequence:

- If there are 2 or more High's in the combination results in a High uncertainty
- If there is one High and one Medium in the combination or if there are two or more Mediums in the combination results in a Medium uncertainty
- If there are more than three Low's in the combination, results in a Low uncertainty

In the scenario, the Decision Maker makes the decision to interact with or engage the IoT device (or not) based on the classification from the combination of uncertainties. We have implemented three types of Decision Maker agents namely a Risky Decision Maker, a Conventional Decision Maker and an Unpredictable Decision Maker. The Conventional Decision Maker as the name suggests exhibits a conventional behavior. It always opts for a decision of 'No' for High and Medium Uncertainty and 'Yes' for Low uncertainty. The Risky Decision Maker agent exhibits a risky behavior. A High and Medium Uncertainty could result in either a 'Yes' or 'No' decision. However, a Low uncertainty always results in a 'Yes' decision. An Unpredictable Decision Maker makes completely arbitrary decisions. The decision for all types of uncertainty classifications i.e. High, Medium and Low could result in either a decision of 'Yes' or 'No'.

Following our previous work in decision making with uncertainty using agent-based simulations [4], all the three types of Decision Makers have an inherent attribute namely the "decision" attribute. Every agent's decision attribute is initialized with a predetermined value at the start of the simulation. This attribute changes in value based on the decision made as per the uncertainty information. During the simulation, we calculate the decision attribute of the Decision Makers each time a decision is made. The decision attribute accounts for the behavior of the Decision Maker and the rules we have set for the classification of uncertainty of information value. For example, if a Conventional Decision Maker's decision are in line with the classification rules then its decision attribute score will increase and vice versa.

The figure below shows the change in value of the Decision Maker's decision attribute each time a decision task is performed.

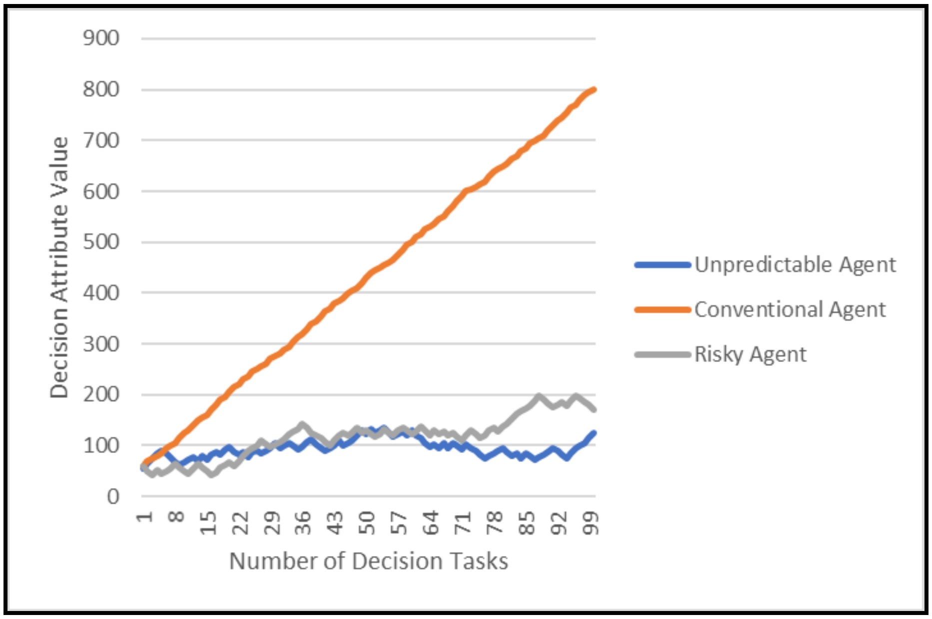

Figure 4: Decision Tasks Results

As depicted in the figure above, we can gauge the performance of various types of Decision Makers in terms of their decision value attribute after performing a certain number of decision tasks. The Conventional Decision Maker fares the best with highest decision attribute value followed by Risky Decision Maker. The Risky Decision Maker is closely followed by the Unpredictable Decision Maker.

6. CONCLUSION

The scenario developed using the SAGE framework gives us preliminary results that help investigate the effects of uncertainty in decision making. Currently in this work, we have looked at general categories of uncertainty sources related to IoT. As this work continues, we will look at different sources of uncertainty, in detail. By clearly defining sources of uncertainty related to IoTs, we can determine which sources contribute to and are valued the most in decision making with respect to the decision maker under specific conditions.

REFERENCES

[1] SAGE, https://www.nrl.navy.mil/itd/imda/products/sage

[2] Gershon, N., "Visualization of an Imperfect World", IEEE Computer Graphics and Applications, 18(4), 43-45, (1998).

[3] https://robotframework.org/

[4] Raglin, A., Metu, S., "Agent Based Simulation of Decision Making with Uncertainty", SPIE Defense + Commercial Sensing, (2019).