

# From Lab to Internship and Back Again: Learning Autonomous Systems through Creating a Research and Development Ecosystem

Trevor Bihl,<sup>†</sup> Todd Jenkins,<sup>†</sup> Chadwick Cox,<sup>‡</sup> Ashley DeMange,<sup>†</sup> Kerry Hill,<sup>†</sup> Edmund Zelnio<sup>†</sup>

<sup>†</sup>Sensors Directorate, Air Force Research Laboratory (AFRL), Wright Patterson AFB, OH 45433

<sup>‡</sup>KeyW Corporation, Dayton, OH

{Trevor.Bihl.2, Todd.Jenkins, Chadwick.Cox.ctr, Ashley.Demange, Kerry.Hill, Edmund.Zelnio}@us.af.mil

## Abstract

As research and development (R&D) in autonomous systems progresses further, more interdisciplinary knowledge is needed from domains as diverse as artificial intelligence (AI), bi-ology, psychology, modeling and simulation (M&S), and robotics. Such R&D efforts are necessarily interdisciplinary in nature and require technical as well as further soft skills of teamwork, communication and integration. In this paper, we introduce a 14 week, summer long internship for developing these skills in undergraduate science and engineering interns through R&D. The internship was designed to be modular and divided into three parts: training, innovation, and application/integration. The end result of the internship was 1) the development of an M&S ecosystem for autonomy concepts, 2) development and robotics testing of reasoning methods through both Bayesian methods and cognitive models of the basal ganglia, and 3) a process for future internships within the modular construct. Through collaboration with full-time professional staff, who actively learned with the interns, this internship incorporates a feedback loop to educate and perform fundamental R&D. Future iterations of this internship can leverage the M&S ecosystem and adapt the modular internship framework to focus on different innovations, learning paradigms, and/or applications.

## Introduction

Autonomous systems promise to reduce workloads and to improve performance by replacing the cognitive functions of people with intelligent software. Robotics are a key technology in many proposed future autonomous systems, from autonomous vehicles to industrial automation and service. Current research involves improving robotic systems to minimize the inputs of human operators by automating tasking. Various perspectives exist in what is/is not autonomy (Bihl et al. 2018) and what technology needs to

be explored to enable autonomy (Floreano and Wood 2015).

Recent work has explored interactive robot laboratories for teaching various artificial intelligence (AI) skills, c.f. (Ruvolo 2017) (Hettlinger and Boutell 2010); but frequently, these courses are highly structured and not directly research related. Summer internships and laboratories in robotics are well documented, c.f. (Grimm et al. 2017) (Burgsteiner et al. 2016) (Guzzi et al. 2018); however, these are highly structured towards specific learning objectives. While open-ended robotics projects have been developed for independent research development, see (Musicant et al. 2017), what is missing is a research-based internship in robotics which aims to both educate the interns and the workforce as well as making Research and Development (R&D) progress in autonomous systems.

Appropriately developed, internships are used as a method of workforce education wherein interns are free to focus entirely on one endeavor and then relay their knowledge back to their mentors. Internships provide various benefits to all parties involved (Coco 2000) (Weible 2009), including evaluating potential employees before a hiring decision and providing experiences to interns while in school. Schools gain further validation of their curriculum and potential employment of their students.

Additionally, workforce development is possible wherein interns and mentors learn side-by-side through collaboration in R&D (Arnold et al. 2008). In such approaches, intern projects provide a useful way to quickly evaluate technologies for further development by the mentoring organization. R&D progress may continue their education in those areas that are determined to be fruitful. Interns, mentors, and their organizations all gain from the experience.

This internship employs a multidisciplinary undergraduate research team to answer various R&D questions in AI,

---

Official works of the US Government are not subject to copyright © 2019.

robotics, and computer science. These included, but were not limited to:

- Can a workforce quickly learn and conduct research in novel autonomy enabled methods inspired by neuroscience?
- How difficult is it for non-experts to quickly learn and apply AI and autonomy methods to solve problems?
- Can a robotics-based framework for testing AI and autonomy technologies and ideas be developed and implemented over a summer?
- Can traditional viewpoints of software design and development, such as agile methods and team effort, be applied by interns to achieve aggressive AI and autonomy goals in a short term? If not, what is the most effective approach?

To answer these questions, the authors developed intern projects which were intended to be simultaneously: 1) independent, 2) complimentary, and 3) integrated; while being constrained to a fourteen week summer program. An additional end goal was the development of a Modeling and Simulation (M&S) ecosystem for testing autonomy concepts.

## Motivations

The motivations for this project are: 1) understanding and teaching a wide set of autonomy, AI, and robotics skills through experiential learning, 2) extending the state-of-the-art in AI through intern-driven and mentor-guided research, 3) integrating multiple intern projects as a team effort to demonstrate incremental autonomy capabilities, and 4) build an ongoing effort of professional development through iterative, feedback interactions between embedded professionals and multiple sets of interns.

## Autonomous Systems Viewpoint

Autonomous machines are expected to be self-reliant and self-tasking. They must complete tasks in unconstrained and poorly characterized situations that were not expected during their development (Bihl et al. 2018). They must make intelligent, informed, and unforced choices. They must have a sense of self that informs their interactions with the environment and with other agents.

Attributes inherent in autonomous systems include: task flexibility, peer flexibility, and cognitive flexibility (Rogers et al. 2018) (Bihl et al. 2018):

- **Task flexibility:** ability of an agent to identify, select, act, and complete multiple tasks to complete an activity
- **Peer flexibility:** refers to an agent’s ability to change from subordinate, peer, and supervisory roles as necessary to complete an activity

- **Cognitive flexibility:** adaptive cognition, an agent’s ability to change decision boundaries, rules, or models.

This internship focused on cognitive and task flexibilities by evaluating and leveraging a wide variety of complementary technologies

Various technologies and components are hypothesized as contributing to autonomy, as conceptualized in Figure 1. These components are conceptually linked to form an OODA (Observe-Orient-Decide-Act) loop (Boyd 2017), which emphasizes the assessment and understanding of human action and reasoning (Behere and Törngren 2016). As discussed in Behere and Törngren (2016), the Observe part of the OODA loop relates to sensing and sensing algorithms; thus, effective sensors and sensor data algorithms are critical to autonomous systems. The Orient and Decide parts relate to reasoning over the sensor data and determining the appropriate courses of action from both the sensed data and past experiences. The Act then corresponds to the actuation of the platform itself. This viewpoint was emphasized in developing projects for this internship.

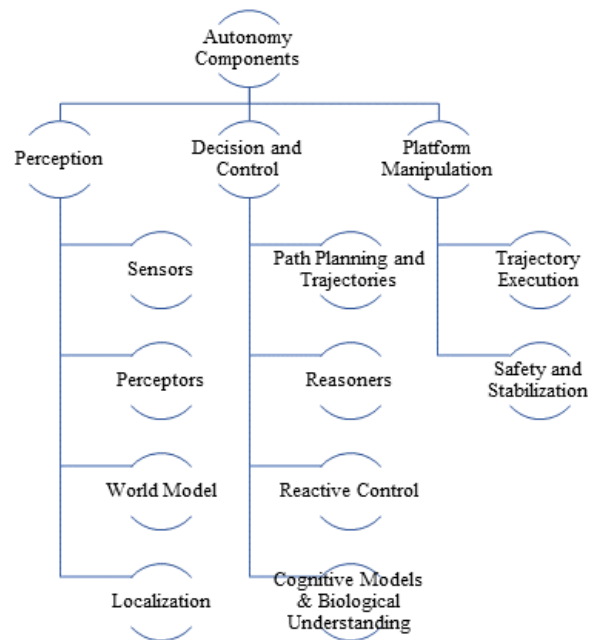


Figure 1. General components and technologies needed for autonomous systems, adapted from (Behere and Törngren 2016).

## Research Motivations

Leveraging the scope of components needed for autonomous systems, as seen in Figure 1, this internship was developed to accomplish a few specific research tasks

- Explore bio-plausible machine learning methods and cognitive models for advanced decision making and autonomous systems

- Develop and integrate an end-to-end M&S ecosystem for testing autonomous systems and components for autonomous systems
- Evaluate the performance of AI and autonomy enabling software on different hardware and processor designs
- Incorporate robotics and integrate developed reasoners and software across the M&S ecosystem.

The M&S ecosystem addresses the realization of the end-to-end process conceptualized in Figure 2, this process contains autonomy and OODA-loop related concepts as discussed in Figure 1. The process shown in Figure 2 creates an M&S environment, including key components of analytics models (abstract but convenient) to physical testing (realistic but costly) (Hill and Miller 2017), for testing of autonomy enabling technologies and methods. The Digital M&S process in Figure 2 is critical for rapidly testing ideas because physical platforms are complex and require consideration of real world nuances. Similarly, feedback of results is critical to aid in both development/improvement of autonomy components along with M&S and testing considerations.

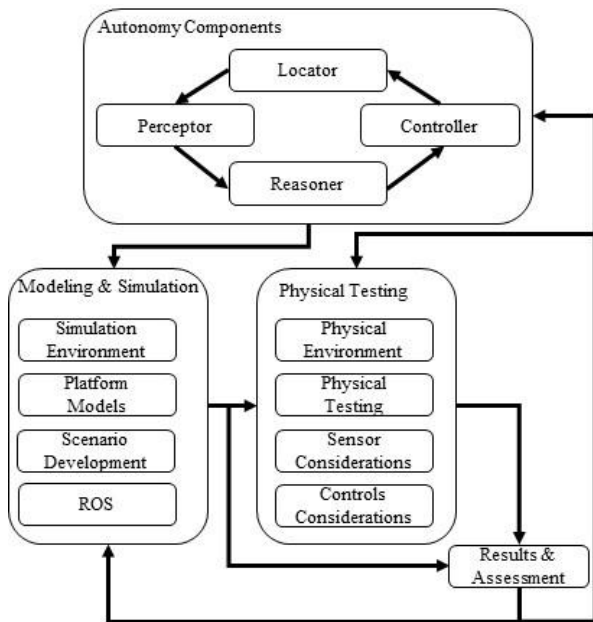


Figure 2. General flowchart of end-to-end digital M&S ecosystem for autonomous system technologies.

Through the process used in Figure 2, components of an autonomous system (reasoners, controllers, perceptrs, locators, and sensors), are employed with hardware or software simulations in an appropriate M&S. The modular nature of ROS (Robot Operating System) facilitates incorporating a variety of instances of these components, and the incremental evaluation of many algorithms. By using ROS compatible M&S environments, algorithms and mod-

ules can be ported to a physical robot with appropriate interfaces to real robot hardware. Existing ROS modules provide two additional advantages, a set of baseline algorithms for comparison to novel algorithms, and a risk reduction whereby baseline algorithms can be used should one intern fail.

For the internship, a TurtleBot2, as seen in Figure 3, was used in both M&S and for physical testing. Beyond typical configurations, the TurtleBot2 was further outfitted with Jetson TX2 boards, a uEye camera (IDS GmbH, Obersulm, Germany), and a Logitech webcam. The internship was then about developing the M&S ecosystem, developing M&S abilities, developing and testing algorithms for reasoning, integrating components, and then operating the robot.



Figure 3. TurtleBot2 demonstration platform.

Reasoning for autonomous systems includes various algorithms, path planning, cognitive models and architectures, and approaches (Dillmann 2004) (Behere and Törn-gren 2016). The primary R&D focus of this internship was the development of a reasoner, which is responsible for decision and control based on sensor data input, situational awareness and memories. Two different approaches for reasoning were considered to enable general comparisons and explore different reasoning constructs. The first approach is a Bayesian Belief Network, created via ProbLog (De Raedt et al. 2007), and the second was a bio-plausible model basal ganglia model created with Artificial Neural Networks (ANNs) in Nengo using theories from the Neural Engineering Framework (NEF) and Semantic Pointer Architecture (SPA) of (Eliasmith 2013) (Bekolay et al. 2014). Nengo, a Python package, incorporates Spiking Neural Networks (SNNs) which are bio-plausible 3rd generation neural networks.

Biological mimics are of interest because, to date, biological systems provide the only proof that truly autonomous agents are possible. The hope is that biological analogies will provide breakthroughs that are needed to solve the tough problems. The work of Sharma et al. (2017) illustrated that NEF can create bio-plausible Bayesian reasoning constructs. So it is instructive to compare and contrast Bayesian solutions, which represent the present-best synthetic approaches to handling uncertain situations, with NEF-based solutions, which attempt to emulate components of biological autonomous agents. Within SPA, a basal ganglia approach was studied because this is the brain mechanism selects actions based on salience (Stewart et al. 2010).

### Pedagogical Motivations

Engineering education is frequently “chalk and talk”, heavily focused towards lectures and traditional pedagogy (Mills and Treagust 2003). When project-based and problem-based learning are considered, it is highly structured and rarely research focused (Mills and Treagust 2003). Similar to the approach of (Manhire et al. 2002) (Ayitey 2018), the internship program diverges from the typical structured pedagogical model and uses physical robotics systems which are of contemporary interest in many domains. Such real world applications naturally overlaps the holistic educational goals of the Accrediting Board for Engineering and Technology’s (ABET) Engineering Criteria 2000 (EC2000) (Manhire et al. 2002).

To facilitate development of novel AI solutions in the M&S ecosystem, an R&D-oriented cycle was incorporated into this internship. R&D can consist of many different high-level tasks and conducting research can thus imply many activities. The concept and process for this internship notably encompasses a variety of high-level tasks, all of which are components of research. The cycle of various research activities is conceptualized in Figure 4, which is extended from various research cycles, e.g. (Hevner 2007) (Sudsawad 2007). A robust research group potentially employs a constant feedback loop to accomplish all these aims. The cycle considered by the authors for this internship included all of these areas:

- Preliminary research and literature reviews
- Innovation of new methods
- Integration of existing methods to new problems
- Optimization of existing methods
- Evaluation and science of testing
- Updating and improvement of methods
- Transitioning and integration of research into a customer’s problem

In this internship, all tasks seen in Figure 4 were considered to some degree. Thus, the interns in this research endeavor were pushing theoretical boundaries, taking existing methods and improving/optimizing, integrating multi-

ple technologies together and evaluating results in both modeling and simulation and with physical robotics.

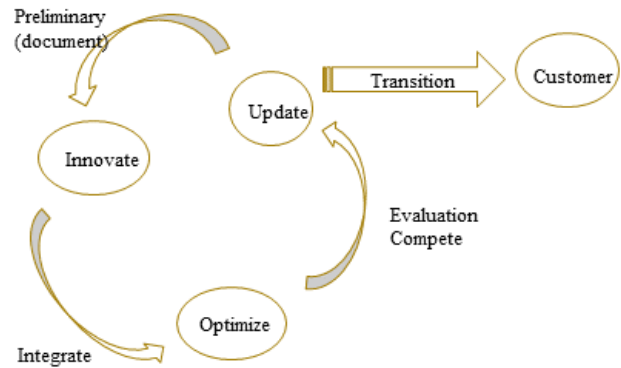


Figure 4. General Cycle of Research, adapted from (Mindrup et al. 2010).

Additionally, the internship employed a structured but open-ended project-based and problem-based process. The final goals and available tools of the project were stated and planned upfront, but failure was a possibility. The interns were taught the important role that risk and failure play in the scientific process. To facilitate success, the mentors employed a research and interaction model as depicted in Figure 5. Figure 5 illustrates how mentors and support staff were frequently in the loop through collaboration, discussion, sampling, and feedback. The model and approach in Figure 5 illustrates how the mentors developed the research plan for the internship prior to the summer and as part of ongoing research. To encapsulate the concepts in Figure 4, internship is divided into three parts: Training, Innovation (R&D), and Integration, as divided in Figure 5.

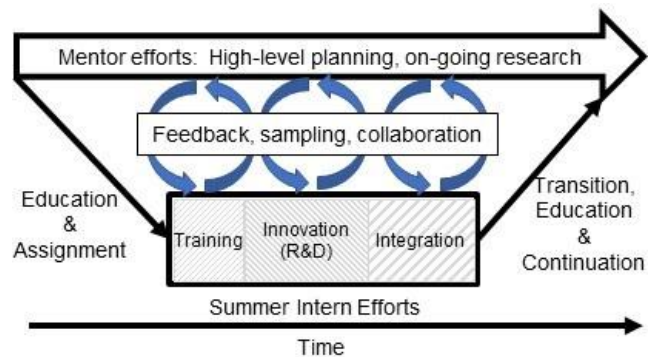


Figure 5. Conceptualization of Mentor-to-Intern interaction and workflow in this internship.

Prior to the start of the summer, the mentors developed the topics and high-level plans/goals. At the start of the summer, the mentors gave preliminary tasks and delivered

lectures and tutorials to the interns. At the completion of training, the interns were then presented with the proof of concept tasks and plan for end of summer integration.

Throughout the summer, the mentors collaborated and interacted daily (in-person or by email) with the interns and guided the research along. To avoid interns becoming overwhelmed, the complete final objectives were slowly discussed and only fully presented at the middle of the summer. However, the mentors later learned that this was not ideal from the intern standpoint since they did not initially know the end goal. At the end of the summer, the interns transitioned their work to the mentors through tutorials, help documentations, briefings, and a poster session. Finally, the work of the interns was transitioned to the mentors for ongoing development.

### Intern Selection and Task Allocation.

Interns were selected from a pool of student applicants for their general background knowledge and expertise in a few areas: computer science, biology and neuroscience, and robotics. Five interns were selected:

- Intern 1. Freshman general science major
- Intern 2. Sophomore Computer Science (CS) major
- Intern 3. Senior Biological Sciences (Bio) major
- Intern 4. Senior Mechanical Engineering (ME) major
- Intern 5. Senior Computer Science (CS) major.

The interns were selected for basic prior skills, but were not expected to have prior experience in autonomous systems, AI, or specific algorithms and software packages planned for use. The mentors ascertained the skills of the interns by reviewing application forms and resumes.

The interns were expected to fill complementary, team-oriented roles. Intern 1 was selected for developing general M&S environments, as well as serving as the integration lead when technologies were extended to a candidate robot platform. Intern 2 was selected for programming abilities to develop autonomous modules for the reasoner. Intern 3 was selected to collaborate with the Intern 2 in developing biologically plausible and biologically inspired algorithmic methods and in preparing literature reviews on cognition and sensing. Intern 4 was selected for experience in robotics, ROS, and programming. Intern 5 was selected for knowledge of hardware and programming to explore porting AI software directly to different processors.

### Necessary Skills and Intern Objectives.

In order to achieve the goals of an autonomy technology testing ecosystem, the interns must develop or facilitate future development of components mentioned in Figure 1. Because all selected interns had limited prior experience in robotics and no prior experience in autonomy, a variety of skills were planned for development. While overall the development of skills and amount of skills developed by interns was similar in average, not all skills were necessary for development by all interns. Thus, the overall learning objectives for the intern team are those seen in Figure 6.

Figure 6 presents the five interns in the internship as columns and the key skills targeted for development over the summer as rows. Circles are used to indicate the focus per the legend at the bottom of the figure.

Considering the terms at use in Figure 6, “Artificial Intelligence” relates to functions that one generally ascribes to people and that are addressed by machine learning, cognitive architectures, sensing algorithms, and general decision logic and control. Beyond being selected due to its maturity and connection to bio-inspired AI solutions, Nengo has seen use on other robotics platforms, thus facilitating a feasible summer project. In addition, one of the mentors recently attended a workshop on the subject.

Because Nengo was selected as the computational framework, gaining Python programming experience was critical. ROS (the Robot Operating System), and general robotics concepts were additionally key learning objectives for integration on the TurtleBot2. Finally, general M&S tools and techniques were necessary skills to be gained to enable testing on simulated robots without risks to the TurtleBot2.

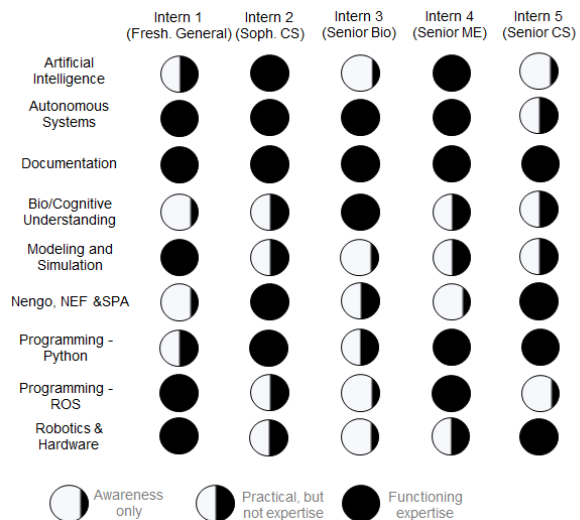


Figure 6. High-level learning objectives and topic area responsibilities for intern team.

## Organization and Projects

Interns were hired through the Autonomy Technology Research (ATR) Center (Arnold et al. 2008), an AFRL Center that provides an environment for collaboration and research across all levels of professionals. It also provided supplies, equipment, and wiki-like documentation abilities to the interns on the first day of work. Significant efforts were made to get the in-terns started as quickly as possible and tutorials on various technologies, e.g. GitHub and Python, were offered throughout the summer.



The ATR Center provides up to fourteen weeks of total internship time with the final week is reserved for end of summer presentations and a poster session on the research. Thus, in general, up to thirteen weeks of research is planned. Realizing this, the internship was divided into three modular parts: training (four weeks), innovation (four weeks), and integration (five weeks). However, fuzzy transitions existed between modules due to natural continuous learning and experimentation.

### Part 1: Training

While many of the skills presented in Figure 6 were gained over the course of the summer, the interns' development was jump started through the use of basic tutorials available for both Nengo and M&S environments.

#### Nengo Tutorials.

To quickly gain an understanding of NEF-SPA, Interns 2, 3, and 5 were initially tasked to read (Eliasmith 2013) while reproducing the fifteen Nengo examples and tutorials on representation, transformation and dynamics, as well as supplemental examples on the basal ganglia. These examples and tutorials followed examples in (Eliasmith 2013) and served to provide the interns with expertise in Nengo, NEF-SPA, autonomy, and general Python programming.

#### Modeling and Simulation Tutorials.

Interns 1 and 4 were tasked to learn the basics of robot M&S with an emphasis towards methods that interface with ROS and Nengo. For this, Intern 1 focused on V-REP (Rohmer Singh, and Freese 2013) and Gazebo (Koenig and Howard 2004), while Intern 4 focused on OpenAMASE (Roberts et al. 2016). V-REP was considered because it is very user friendly and has seen past use in Nengo applications (Nowak and Stewart 2017); Gazebo was considered because it has strong ROS connections and could facilitate directly porting software to the real TurtleBot2. OpenAMASE was considered because it is a general robotics simulation environment which has emphasis to airborne robots.

This part of the project began by examining built-in models to learn control and tasking of platforms as well as the code and message structure. From here, Intern 1 considered open source TurtleBot2 models, e.g. (Silliman 2015) (Lee 2013), as depicted in in Figure 7. Meanwhile, it was decided that the built-in models were sufficient for Intern 4's objectives of developing reasoners.

Final objectives in part of the project included understanding M&S, demonstrating the ability to add models to the M&S environments, and adding and editing code to interact with simulated sensors. Additionally, an understanding of ROS and how it operates was key in this phase to facilitate future porting of software to actual platforms. At the end of this part, the interns additionally documented the use and tradeoffs across the M&S environments.

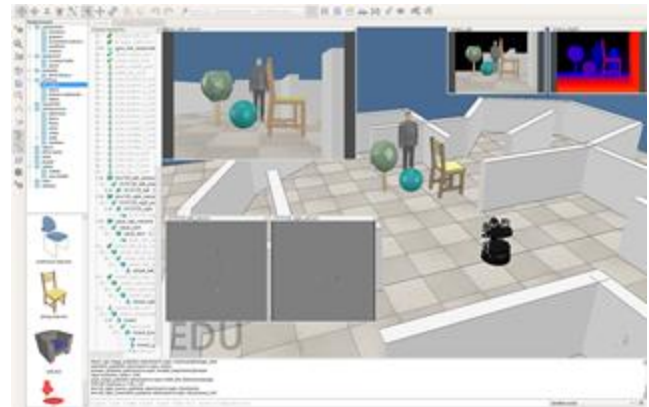


Figure 7. Example V-REP interface with a simulation of the TurtleBot2 demonstration platform in a simulated environment along with simulated sensor feedback.

### Part 2: Innovation, Research and Development

Prior to deploying the concept on physical platforms, or even in M&S, a flow chart and proof of concept idea was created in collaboration with the intern team. The end result, seen in Figure 8, facilitates all of the autonomy-related components discussed in Figures 1 and 2. Development of software used Agile Software Development concepts and involved evolutionary development of code and continual improvement as lessons were learned.

The general concept in Figure 8 starts with perception, with the outputs sent to the reasoner. The reasoner then decides the platform's course of action, which is then sent to the navigation and control components. Since the platform is interacting with a world (real or M&S), its movement triggers changes in its perception and the loop repeats.

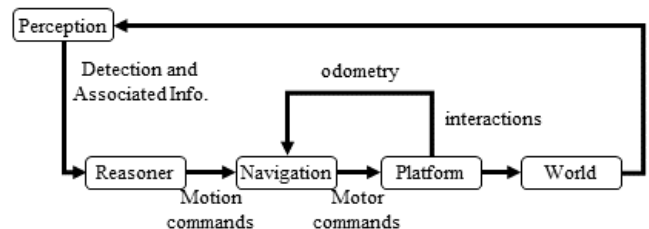


Figure 8. Developed and Employed General Flowchart of Autonomous Systems Operation.

The proof of concept development included extending the foundations learned in part 1 to developing algorithmic reasoners to make decisions for a robot given sensor data input and situational awareness. The intern team was given a great degree of latitude in developing the reasoner, with constant feedback, communication, and collaboration with mentors employed in this stage to ensure the result was usable and potentially useful.

### Reasoner Development.

For the reasoner development, as discussed earlier, both a Bayesian Network (via ProbLog), developed by Intern 4, and a basal ganglia (via Nengo), developed by Interns 2 and 3, were considered. Both approaches were created in Python with goals of similarity in inputs and response.

Intern 3 employed their background in biology to facilitate appropriately structuring a computer generated basal ganglia and implement it into a robot. Additionally, Intern 3 realized that the Nengo basal ganglia model does not include other associated structures, like the substantia nigra and the amygdala. Leveraging this knowledgebase, amygdala-like functionality was incorporated by consider emotions and learning mechanisms with a reasoner developed to have “Intentions” and “Moods.” In the intern developed construct, detected objects were associated with intentions and the collected results of observations change the mood of the robot. Using, and extrapolating from concepts found in literature c.f. (Cox and Krichmar 2009), high level intentions were selected to be “Feed,” “Flee,” and “Explore.” From these ideas, general “Moods” for the module were considered as “Hunger,” “Fear,” and “Curiosity.”

### Hardware Experimentation.

Intern 1 began collaborating with Intern 4 during this part to learn functional ROS abilities and begin working on the navigation and control part of the feedback loop in Figure 8 with the TurtleBot2. Additionally, Intern 4 was primarily tasked in developing a perceptor for the robot. In this process, both interns learned to overcome unexpected integration issues and real world experimentation issues when selecting and incorporating a TensorFlow-based (Abadi, et al., 2016) convolutional neural network (CNN) trained on the COCO dataset.

During this part, Intern 5 was considering hardware tradeoff studies when running both CNNs in Python and SNNs in Nengo on different hardware configurations. Integration issues encompassed the majority of this task with learning how to run Nengo SNNs on commercial off-the-shelf (COTS) device architectures, e.g. FPAs, GPUs, and CPUs. To facilitate understanding of the computational complexity of SNNs versus CNNs, and other 2nd generation ANNs, the Nengo neuron structures assembled by Intern 5 were furthermore similar in size and complexity as those developed by Interns 2 and 3.

### Part 3: Integration, Application, and Testing

In Part 3, the groups came together to take the reasoner and interface it within the M&S environment and the Turtlebot. Challenges here were integrating the Python developed reasoner with ROS and then appropriately handling settings to provide for a realistic sample and decision rate. Appropriate behavior of the robot given simulated sensor data input was considered. With efficacy tested in M&S,

the methods were applied to the real TurtleBot2. It was hoped that a considerable experimentation and testing would be possible; however, and despite heavy mentor collaboration, knowledge gaps and integration issues occupied a majority of the team’s time, thus only limited testing was possible.

Despite setbacks, real operational testing showed efficacy of the reasoner and the entire M&S ecosystem. An example of one sample from real operational testing of the Turtlebot is given in Figure 9 which shows output from the perceptor being fed into the reasoner to determine motion commands and change the mood of the Turtlebot. Here probabilities are from the perception piece as classification confidence values. The reasoner (Nengo or ProbLog based) then determines the intention of the robot given the detection and its current mood. Based on the intention, a command is given to move the robot appropriately.

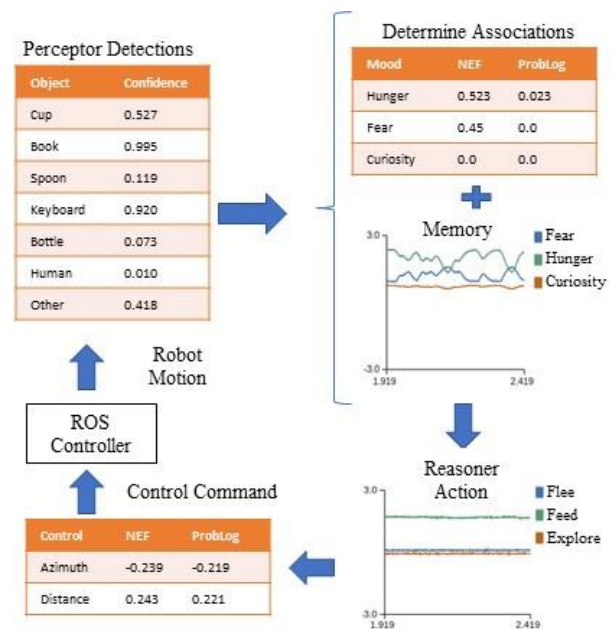


Figure 9. Example output from perceptor, with example object classes. Reasoner gives intents, which result in control outputs and moods (which are fed back to the reasoner).

### Conclusions

Through an internship format, the authors were able to conduct R&D in autonomous systems. In a fourteen week internship format, a novice team was able to demonstrate the ability to learn autonomy, AI, and neuroscience concepts and apply these methods to solve novel problems. The intern team was highly interdisciplinary and included five undergraduates, with backgrounds ranging from computer science to biology. Mentors facilitated both education and integration and the result was a set of five projects that were simultaneously: 1) independent, 2) complimentary, and 3) integrated. Combined together, the individual intern

projects further yielded a digital M&S ecosystem for testing autonomy concepts through both virtual and physical robotics testing.

In creating the M&S ecosystem, proof of concept development of AI systems based on both Bayesian networks and basal ganglia models were developed and analyzed for real-time reasoning about the environment. Through this diverse team background, bio-plausible architectures were developed and knowledge of biology was used to influence designs. The end results included an intern team trained in autonomous technology, improved workforce understanding of technology pieces, and a starting point for future autonomous research endeavors.

### Future Internship Iterations and Extensions

As constructed, and described in Figure 5, this internship model is applicable to future iterations due to its modular nature. Additionally, others can leverage the overall internship design to develop their own M&S ecosystem and use the internship model, Figure 5, to train and enable interns to perform in R&D for autonomous systems.

Future iterations of the internship model could see the training component remain the same, while the “Innovation (R&D)” and “Integration” pieces would differ and focus on a different application. While the general M&S ecosystem of Figure 2 would not change in future iterations, different constituent components are envisioned. Such work could focus on entirely different AI approaches, applications, alternative platforms, additional sensors, or on further advances within the current M&S ecosystem.

### Future Technical Work

While the end result showed the efficacy of the ecosystem and a general level of autonomous functionality, various technical issues must be corrected to facilitate future work. Primarily this involves vetting the quality of ancillary software modules prior to use. To avoid future interns frequently fixing problems or adding lower-level software capabilities, mentors will carefully evaluate and test ancillary software modules, e.g. perceptors, prior to intern arrival. To this end, the ROS framework is beneficial because it offers many modular capabilities.

Beyond such logical near term items, more medium term items for future interns include using the developed M&S ecosystem to develop and test the reasoners further, and vetting the mood and intent construct developed by this intern team. Additional technical work can involve considering different reasoner architectures within the NEF-SPA and ProbLog constructs, as well as through other, yet to be determined, reasoning methods. Additionally, work will be needed in designing and testing appropriate scenarios, developing and evaluating metrics for autonomous system

performance, and considering different approaches for autonomous system reasoning.

### Lessons Learned

A variety of lessons were learned in this project, both technical and non-technical. Technical lessons include realizing that additional time spent on planning and data wrangling by the mentors was needed, as was better preparation of ancillary tools. An experimental approach should also be prepared to direct students towards appropriate test methods and metrics. Test methods and metrics for autonomous systems are themselves a very open area of research, see (Bihl et al. 2018), so either a mentor or intern could be assigned to this important research endeavor.

On the non-technical side, messaging of goals and roles needed improvement. Interns expressed a desire to have been more informed of the end goals; however, the mentors decided at the start to slowly deliver this information to avoid overwhelming the team. Interpersonal issues were also found and interns had difficulty expressing their frustrations with mentors and learning to accept criticism. Thus, future iterations of this internship will focus on more careful intern selection, revising goals as appropriate with interns, and team development mechanisms to provide cohesion between intern teams and between mentors and interns (Knight, 2006).

Additionally, the authors found that the holistic educational goals of ABET EC2000, see (Manhire et al. 2002) for details, are not always realized in undergraduate education and thus explanations and team forming exercises are further beneficial in future intern iterations from an educational perspective. The authors also believe that further dialog between academia and the R&D community can be beneficial to enable better educated graduates who are familiar with in-demand concepts and technologies.

### References

- Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; and Kudlur, M. 2016. Tensorflow: a system for large-scale machine learning. *OSDI*, 265-283.
- Arnold, G.; Ross, T.; Westerkamp, L.; Carin, L.; and Moses, R. 2008. The ATR Center and ATRPedia. *SPIE Proceedings*.
- Ayitey, K. 2018. *Exploring the Global Competence of Students Through the Global Leadership Center: A Qualitative Case Study on the Lived Experience of Graduates*. Ohio University.
- Behere, S., and Törngren, M. 2016. A functional reference architecture for autonomous driving. *Information and Software Technology*, 73, 136-150.
- Bekolay, T.; Bergstra, J.; Hunsberger, E.; DeWolf, T.; Stewart, T.C.; Rasmussen, D.; Choo, X.; Voelker, A.; and Eliasmith, C. 2014. Nengo: a Python tool for building large-scale functional brain models. *Frontiers in neuroinformatics*, 48.



- Bihl, T.; Cox, C.; and Jenkins, T. 2018. Finding common ground by unifying autonomy indices to understand needed capabilities. *SPIE Defense and Commercial Sensing*.
- Boyd, J. 2017. *A discourse on winning and losing*. Air University Press.
- Burgsteiner, H.; Kandhofer, M.; and Steinbauer, G. 2016. IRobot: Teaching the Basics of Artificial Intelligence in High Schools. *AAAI*, 4126-4127.
- Coco, M. 2000. Internships: A try before you buy arrangement. *SAM Advanced Management Journal*, 65(2), 41.
- Cox, B., and Krichmar, J. 2009. Neuromodulation as a robot controller. *IEEE Robotics & Automation Magazine*, 16(3), 72 - 80.
- De Raedt, L.; Kimmig, A.; and Toivonen, H. 2007. ProbLog: A probabilistic Prolog and its application in link discovery. *International Joint Conference on Artificial Intelligence (IJCAI)*, 2468-2473.
- Dillmann, R. 2004. Teaching and learning of robot tasks via observation of human performance. *Robotics and Autonomous Systems*, 47(2-3), 109-116.
- Eliasmith, C. 2013. *How to build a brain: A neural architecture for biological cognition*. Oxford University Press.
- Floreano, D., and Wood, R. J. 2015. Science, technology and the future of small autonomous drones. *Nature*, 521(7553), 460-466.
- Grimm, C.; Lyman-Holt, A.; and Smart, W. 2017. A Summer Research Experience in Robotics. *Educational Advances in Artificial Intelligence*, 4775-4780.
- Guzzi, J.; Giusti, A.; Di Caro, G. A.; and Gambardella, L. M. 2018. Mighty Thymio for University-level Educational Robotics. *AAAI*, 7952-7953.
- Hettlinger, A., and Boutell, M. R. 2010. A Simulator for Teaching Robotics Programming Using the iRobot Create. *AAAI*, 1915-1916.
- Hevner, A. 2007. A three cycle view of design science research. *Scandinavian journal of information systems*, 19(2), 87-92.
- Hill, R., and Miller, J. 2017. A history of United States military simulation. *Winter Simulation Conference (WSC)*, 346-364.
- Knight, P. 2006. *Small, short duration technical team dynamics* (No. S-RES-024-XXX-R2-04). Ft. Belvoir, VA: Defense Acquisition University.
- Koenig, N., and Howard, A. 2004. Design and use paradigms for Gazebo, an open-source multi-robot simulator. *IROS*, 2149-2154.
- Lee, J. 2013. *turtlebot\_description*. Retrieved from ROS.org: [http://wiki.ros.org/turtlebot\\_description](http://wiki.ros.org/turtlebot_description)
- Manhire, B.; Emery, G. A.; Mould, D. H.; and Noland, C. M. 2002. Ohio University's Global Learning Community. *American Society for Engineering Education Conference and Exposition*, 1-8.
- Mills, J., and Treagust, D. 2003. Engineering education—Is problem-based or project-based learning the answer. *Australasian journal of engineering education*, 3(2), 2-16.
- Mindrup, F.; Bauer, K.; Bihl, T.; and Williams, J. 2010. AFIT/ENS Sensor Fusion Lab Advanced Research in Automatic Target Recognition. *Military Operations Research Society (MORS) Symposium*.
- Musicant, D. R.; Laddha, A.; and Choi, T. 2017. Open-Ended Robotics Exploration Projects for Budding Researchers. *AAAI*, 4792-4797.
- Nowak, P., and Stewart, T. 2017. A Spiking Model of Desert Ant Navigation Along a Habitual Route. *23rd International Conference on Soft Computing*, 211-222.
- Roberts, M.; Hiatt, L.; Coman, A.; Choi, D.; Johnson, B.; and Aha, D. 2016. ActorSim, A toolkit for studying crossdisciplinary challenges in autonomy. *Cross-Disciplinary Challenges for Autonomous Systems: Papers from the AAAI Fall Symposium*.
- Rogers, S.; Peterson, G. L.; and Mendenhall, M. J. 2018. *Autonomy Capability Team 3 (ACT3): Development Goal*. White Paper.
- Rohmer, E.; Singh, S.; and Freese, M. 2013. V-REP: A versatile and scalable robot simulation framework. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 1321-1326.
- Ruvolo, P. 2017. Dude, Where's My Robot?: A Localization Challenge for Undergraduate Robotics. *AAAI*, 4792-4802.
- Sharma, S.; Voelker, A.; and Eliasmith, C. 2017. A Spiking Neural Bayesian Model of Life Span Inference. *Proceedings of the 39th Annual Conference of the Cognitive Science Society*, 3131-3136.
- Silliman, M. W. 2015. *We Want You to Learn TurtleBot in Simulation!* Retrieved from Learn TurtleBot and ROS: <http://learn.turtlebot.com/2015/02/03/1/>
- Stewart, T.; Choo, X.; and Eliasmith, C. 2010. Dynamic Behaviour of a Spiking Model of Action Selection in the Basal Ganglia. *10th International Conference on Cognitive Modeling*, 235-240.
- Sudsawad, P. 2007. *Knowledge translation: Introduction to models, strategies, and measures*. 2007. Austin, TX: Southwest Educational Development Laboratory, National Centre for the Dissemination of Disability Research.
- Weible, R. 2009. Are universities reaping the available benefits internship programs offer? *Journal of Education for Business*, 85(2), 59-63.