

ABSTRACT

Title of Thesis:

DEVELOPMENT OF LOW-COST
AUTONOMOUS RESEARCH SYSTEMS

Logan M. Saar, Master of Science, 2023

Thesis directed by:

Professor Ichiro Takeuchi, Materials Science and
Engineering

A central challenge of materials discovery for improved technologies arises from the increasing compositional, processing, and structural complexity involved when synthesizing hitherto unexplored material systems. Traditional Edisonian and combinatorial high-throughput methods have not been able to keep up with the exponential growth in potential materials and relevant property metrics. Autonomous operated Self-Driving Labs (SDLs) - guided by the optimal experiment design sub-field of machine learning, known as active learning - have arisen as promising candidates for intelligently searching these high-dimensional search spaces. In the fields of biology, pharmacology, and chemistry, these SDLs have allowed for expedited experimental discovery of new drugs, catalysts, and more. However, in material science, highly specialized workflows and bespoke robotics have limited the impact of SDLs and contributed to their exorbitant costs. In order to equip the next generation workforce of scientists and advanced manufacturers with the skills needed to coexist with, improve, and understand the benefits and limitations of these autonomous systems, a low-cost and modular SDL must be available to them. This thesis describes the development of such a system and its implementation in an undergraduate and graduate machine learning for materials science course. The low-cost SDL system developed is shown to be affordable for primary through graduate level adoption, and provides a hands-on method for simultaneously teaching active learning, robotics, measurement

science, programming, and teamwork: all necessary skills for an autonomous compatible workforce. A novel hypothesis generation and validation active learning scheme is also demonstrated in the discovery of simple composition/acidity relationships.

DEVELOPMENT OF LOW-COST AUTONOMOUS RESEARCH SYSTEMS

by

Logan M. Saar

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2023

Advisory Committee

Professor Ichiro Takeuchi, Chair
Professor Ji-Cheng Zhao
Dr. Aaron Gilad Kusne

©Copyright by
Logan M. Saar
2023

Acknowledgements

I am extremely lucky to have had so many encouraging and helpful people in my life over the past 2 and a half years, all of whom deserve appreciation for their support.

To my research advisor, Dr. Ichiro Takeuchi, I am truly thankful for the opportunity I have had to receive your guidance and assistance throughout my research. It has been a pleasure to be mentored by you and I am grateful for the chance you gave me to implement my research in the classroom as an educational tool. To my other advisor, Dr. Gilad Kusne, I am equally thankful for your guidance and patience when helping me to overcome the hurdles in this project. To both of you, your constructive communication, patience, and friendliness has been a stellar example to me of not only how to conduct research, but also how to treat others well.

Thank you to my entire research group and other students I have had the pleasure to work with during my time at university: you have made me feel welcome and helped me grow as an individual by sharing your knowledge and offering your support. Thank you to Haotong for the countless hours of helping me troubleshoot code and help students; to Felix for the valuable assistance in facilitating the use of the robots in the classroom, to Alex, whose introduction to the project and previous work set me up well for success; and the many others who have helped me in small ways - I hope to return the favor with my own support.

I would like to give a big thanks to my family and my friends, who have always given me unconditional support, love, and loyalty. You have all carried me through this busy time in my life and always made it easier with your attention and presence.

Thank you to my defense committee members for being so generous with their time. To Dr. JC Zhao, your support of this project and my education has been a true blessing, and I am extremely grateful for your confidence in me. To Adaire Parker, Dr. Isabel Lloyd, and the entire

MSE department at UMD: thank you for presenting me with all the opportunities to give back to the program and for your assistance in helping me navigate my coursework. Finally, a big thanks to the National Institute of Standards and Technology for their financial support during the Summer Undergraduate Research Fellowship in 2021.

Table of Contents

Acknowledgements	ii
Table of Contents	iv
List of Tables	vi
List of Figures	vii
Chapter 1: Introduction	1
1.1 Perspective & Motivation	1
1.2 Thesis Overview	3
1.3 Background	5
1.3.1 Active Learning Schema.....	5
1.3.2 Relevant ML Concepts	9
1.3.3 Autonomous Experimentation in Materials Science.....	12
1.3.4 Educating an SDL Compatible Workforce	14
1.4 Thesis Outline: Ch.2 - Ch.4 (Development and Operation)	24
Chapter 2: Systems Development	25
2.1 Design Principles	25
2.2 Mechanical Design.....	28
2.3 System Construction & Preparation.....	31
2.4 Connecting to & Calibrating LEGOLAS.....	39
2.5 Developmental Stages.....	43
Chapter 3: Educational Implementation	50
3.1 Henderson-Hasselbalch Exercise.....	50
3.1.1 Introductory Exercises	53
3.1.2 Autonomous Closed Loop Exercise.....	57
3.2 Class Implementations	62
3.2.1 Fall 2021	63
3.2.2 Fall 2022	63
3.2.3 Lessons Learned.....	64
Chapter 4: Autonomous Model Exploration	67
4.1 Bayesian Optimization.....	67
4.2 Hypothesis Validation Objective	70
4.2.1 Generating and Evaluating Candidate Hypotheses.....	73
4.2.2 Informational Entropy Acquisition Function.....	76
Chapter 5: Future Work and Conclusions	81
5.1 Future Work & Scope	81
5.2.1 Alternate Educational Exercises	81
5.2.2 System Modifications	82
5.2 Conclusions.....	83

Appendices.....	84
I: Associated Media	84
Bibliography	85

List of Tables

Table 1.1 Summary of some recent low-cost SDL platforms	19
Table 2.1 Cost Breakdown for a single LEGOLAS robot	32
Table 2.2 Summary of FDM printed parts	33

List of Figures

Figure 1.1 Picture of LEGOLAS & its use in the Fall '21	5
<i>Machine Learning for Materials Science</i> course at UMD	
Figure 1.2 Pictorial representation of Bayes Theorem for iteratively updating our beliefs	11
Figure 1.3 Three target foundational pillars for educational development identified by an MGI working group	15
Figure 1.4 Images of low-cost SDLs from <i>Table 1.1</i>	20
Figure 1.5 Simplified pictorial representation for an SDL such as LEGOLAS	23
Figure 2.1 Experimental components on the trolley	29
Figure 2.2 Camera attached to the bottom of the Trolley	30
Figure 2.3 All 3D printed parts needed to build one LEGOLAS robot	33
Figure 2.4 Fully assembled stand	34
Figure 2.5 Fully assembled bridge	35
Figure 2.6 Assembled trolley resting on the bridge	36
Figure 2.7 Sample space setup for the Henderson-Hasselbalch pH study	38
Figure 2.8 Location of relevant wires and electronic devices for LEGOLAS	39
Figure 2.9 GUI calibration window and liquid-volume/ gear-step calibration using a mg digital scale	41

Figure 2.10 Example usage of fundamental movement functions and a synthesis/measurement loop	42
Figure 2.11 Liquid handling robot that inspired the LEGOLAS design	43
Figure 2.12 LEGOLAS (<i>Generation I</i>)	44
Figure 2.13 LEGOLAS (<i>Generation II</i>)	45
Figure 2.14 Classroom setup for Fall '21 implementation	46
Figure 2.15 LEGOLAS (<i>Generation III</i>)	47
Figure 2.16 LEGOLAS (<i>Generation IV</i>)	48
Figure 2.17 Students working on LEGOLAS exercises during the Fall '22 implementation	49
Figure 3.1 The Henderson-Hasselbalch equation	51
Figure 3.2 Percent error in the Henderson-Hasselbalch Equation as a function of sodium hydroxide concentration	52
Figure 3.3 Gaussian process demonstrated for 5 samples Within a composition range of %acid = [5-95]	56
Figure 3.4 Gaussian process evolution (RBF Kernel) for First 5 experimental samples with an exploration-based CO	61
Figure 4.1 Sampling of potential models across the prior distribution determined by the kernel	69
Figure 4.2 Probabilistic interpretation of the GP for uncertainty quantification and propagation	69
Figure 4.3 Effect of new data on GP structure and	70

a purely exploration-based acquisition function	
Figure 4.4 Process-flow for experimentation, hypothesis generation, and evaluation	75
Figure 4.5 Process-flow for creating GMMs over sample space	78
Figure 4.6 Using the informational entropy metric to select the next composition	80
Figure 5.1 Camera images of sample wells from the color mixing study	81
Figure 5.2 Analog aqueous conductivity probe, spectrophotometer, and heating element	82

Chapter 1: Introduction

1.1 Perspective & Motivation

In the past 50 years, *automated* systems - systems that perform tasks with little or no human control - revolutionized the manner in which society operated [1]. These new technologies increased productivity and efficiency by automating repetitive and precision based tasks; tasks for which humans were more poorly suited [2]. These automated systems, created by and *for* humans, greatly improved society's ability to produce on a large scale [1,2]. Inherent to this revolution was a shift in the needs of the workplace. On one hand, there was a new demand for people to develop and improve the automated systems themselves [3]. On the other hand, there was an increase in demand for workers - humans - who could tend to, interact with, and understand both the hardware and software elements of the automated systems [3]. At all levels of education, classes that taught the fundamentals of automated systems - from computer programming to robotics - began receiving a greater degree of emphasis [1,3]. These were all efforts to equip the next workforce with skills necessary to coexist with automated systems and thrive in an automated world.

For materials science, this automated revolution expanded the ability to explore and experiment with a larger space of compositions, processing parameters, and performance metrics [4]. In large part, the substantial successes of combinatorial and high-throughput experimentation (CHT) were made possible by automated systems that accelerated the synthesis and characterization steps [4]. However, in automated systems, the “lead scientists” - the experiment directing field-specific human experts are never supplanted by the robotic task-performing machines [4]. Incapable of straying beyond the means of their typical tasks - even if preprogrammed to act differently in certain situations - the automated systems could not

take on the intuition based activity of *understanding* the world through the method of scientific process: observation, hypothesis, and experiment [5,6]. At most, they played a *role* in this process, as a tool which could facilitate and accelerate experiments.

In the past few decades, with the advent of machine learning (ML) and the broader field of artificial intelligence (AI), it has become possible to merge these new technologies with automated robotics into what are known as *autonomous* systems. Capable of *approximating* human intelligence - mainly the faculty of learning and adapting - these systems were endowed with adaptive algorithms for responding to varying stimuli in a non-preprogrammed, non-predetermined fashion. To what degree AI & ML constitutes intelligence with respect to our human definition of the word is already a subject of much debate, but what is clear - from their widespread adoption (albeit still mostly in development stage) and rising popularity - is that they are going to play a large role in the world of the future [1,7-9]. Self-driving cars, autonomous factories, autonomous drones, the internet-of-things: all examples of a data-driven world in which machines have begun to usurp *certain* previously human-performed decision making tasks [10,11]. The pervading question, however, is to what degree could these autonomous systems function in roles where the workflow is not only highly complex and dynamic, but also not clearly defined or understood by humans themselves [5,12-14]. How may an autonomous system perform within a profession in which the role can be ambiguous, dependent on insight, and itself a method of discovery, say, as that of a “lead scientist?” [5,7,8,12,15,16].

To consider this question, as well as work to better illuminate what a “lead scientist” really does, is one of the essential steps in preparing the ground for autonomous experimentation (AE) as a fruitful field for cultivation [5,15,17]. A second related question that arises is: what

role will *humans* continue to play in science, and thereby what skills & fundamentals will be useful for them to cultivate to continue thriving in this changing world? [18-21] This thesis attempts to approach and begin to answer both of these questions. In the background section, the fundamentals of ML, AI, and *active learning* will be discussed in terms of their implementation in AE. Next, the merit of AE will be discussed with respect to prior successes and challenges within the field of Materials Science and Engineering (MSE). We present a LEGO-based low-cost autonomous scientist as an embodiment of an autonomous science kit to be used for educating the next generation workforce. The requirements for an autonomous compatible workforce of *human* researchers will be discussed, focusing on the skills needed and methods of teaching those skills. Additionally, the role of a “lead scientist” and the ability to potentially encapsulate the scientific process of hypothesis generation and testing within an AE system’s “intelligence” will be considered.

1.2 Thesis Overview

This thesis describes my development of low-cost robotic systems for closed-loop AE. These systems are designed to be affordable, modular, and simple to operate to ensure usability at all educational levels. They are composed of inexpensive components, including LEGOs, to allow for the affordability and modularity desired. From here on out, they will be referred to as LEGOLAS: LEGO based Low-cost Autonomous Scientist(s) (*Figure 1.1*).

I began development of LEGOLAS during the Summer Undergraduate Research Fellowship (SURF) program at the National Institute of Standards and Technology (NIST) in 2020. The first task was building the robot and producing a live-run with a purely exploration based campaign objective (*Section 4.1*). Some figures shown in this section are from the

resulting colloquium presentation provided at the end of that Summer [22]. I was then able to further develop the Autonomous Model Exploration modules (*Section 4.2*) while working as an undergraduate research assistant in the ML group at UMD (co-led by Dr. Ichiro Takeuchi at UMD and Dr. Gilad Kusne at NIST). Many of the figures in *Sections 4.1 & 4.2* are from a presentation I gave in December of 2021 at the Materials Research Society (MRS) Symposium on Accelerating Experimental Materials Research with Machine Learning [23]. We were then also able to submit a publication on this work to MRS Bulletin in May of 2022 (Published November 2022) [24]. Further publications on the use of LEGOLAS in the classroom for solid-state materials research, and multi-objective studies, are expected in the coming months and years.

Chapter 2 describes the design principles and methods I used in determining how to construct, operate, and interface with LEGOLAS. Chapter 3 then reviews the use of LEGOLAS in two undergraduate / graduate courses for *Machine Learning for Materials Science* (ENMA 437/637) offered at the University of Maryland (UMD) (*Figure 1.1*). In these classes, I was the lead teaching fellow and designed the exercises for completion, which served as the final project for students.

Chapter 4 discusses the use of LEGOLAS to perform novel on-the-fly hypothesis generation and validation using symbolic regression and an acquisition function rooted in the use of informational entropy as a metric of uncertainty [25,26]. The process of active learning development displayed in Chapter 4 is meant to be a testament to the ability of LEGOLAS to facilitate a creative and responsive learning environment in which the encoding of patterns of human “intelligence” into AE systems is more easily achieved. Chapter 5 talks about the future for LEGOLAS and the educational opportunities it provides.

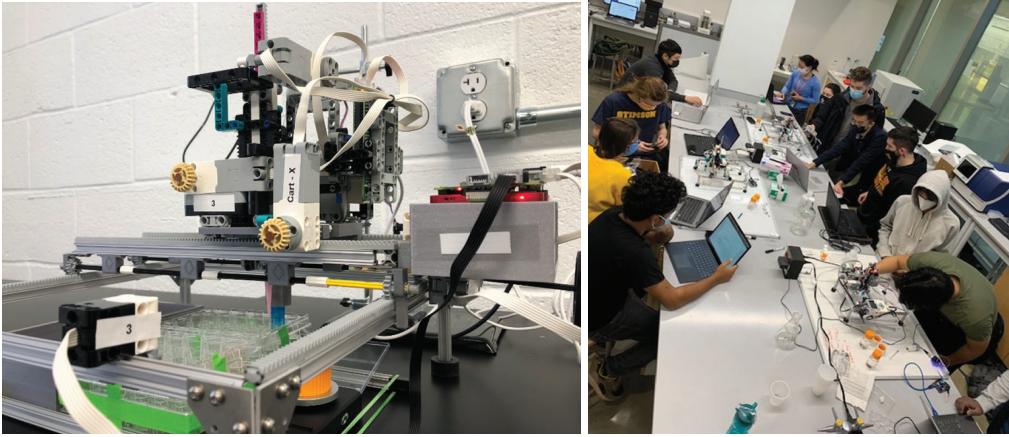


Figure 1.1: (left) Picture of LEGOLAS & (right) its use in the Fall '21 *Machine Learning for Materials Science* course at UMD

1.3 Background

1.3.1 Active Learning Schema

Active Learning is itself a sub-field of ML, and its application in AE systems is what distinguishes them from the rigid, prescriptive behaviors of purely *automated* systems. AE systems are flexible and capable of guiding and altering experimental design - that is, the choice of experimental prerequisites (composition, processing parameters) and/or characterization schemes (means of collecting feedback/data) - to achieve some user defined “goal” [27]. These goals, known as campaign objectives (CO), may be divided into roughly 3 main categories: exploration, exploitation, & mechanistic study [27]. These COs will be described in more details later but briefly:

- an *exploration* CO aims to “explore” a large amount of sample space, prioritizing the investigation of areas in which we have little information (i.e, the *unknown*),
- an *exploitation* CO - also known as an optimization CO - aims usually to maximize or minimize some response metric (feedback or performance measure) and thereby

investigate areas for which we believe the optimal metric *could* be found (i.e use the *known* to make a prediction),

- and a *mechanistic study* CO aims to either determine or attribute an underlying functional form to the data - a relation between input parameters and output values - with the highest possible *confidence*, always considering that this *form* is selected amongst a sea of other potential descriptive forms,

In reality, most active learning schemas are a combination of these COs, and will implement them together or in succession depending on the nature of the experiment. As an example, in a hypothetical materials “discovery” experiment, it may be beneficial to first *explore* the composition space, conducting initial experiments into unknown areas of sample space, and then - once some of the sample space is known - shift to an *exploitation* CO in which subsequent experiments search at or nearby established maxima/minima. In this way, the active learning schema provides an “intelligent” means of searching through the sample space, potentially avoiding the headache of investigating all samples (i.e. exploring the entire time), and additionally avoiding the pitfall of trying to optimize the property when nothing is known (i.e. “shooting in the dark,” or exploiting from the start). This idea of alternating between COs in an active learning schema is known as “scheduling.” It is important to note that in all these COs, we are typically basing our decision making criteria on some idea of the assumed *value* of running a certain experiment. The various methods for quantifying this value and using it for experimental guidance are at the heart of active learning, and will be elucidated in more detail when *acquisition functions* are discussed.

When active learning is implemented in AE towards the achievement of these CO's, it is often in pursuit of a reduced experimental cost. This "reduced cost" can constitute several meanings depending on the constraints and nature of the specific experiment:

- (1) acceleration of the process due to a lack of funds required to carry out experiments or time,
- (2) reduction in number of experiments due to a lack of funds, time, or resources,
- (3) reduction in number of experiments due to a prohibitively large sample space

Reasons (1) & (2) are quite straightforward when considering costly and time-restricted studies - for example, those involving synchrotron beam time or expensive constituents [28] - and reason (1) is to some degree satisfied partially by *automated* experimentation systems (for example, those of combinatorial experimentation studies) since they can reduce sample transfer time, human error, and other associated slowdowns with human-executed processes [4,12,29-31]. Reason (3) arises from the exponential growth in the number of "combinations" of possible experimental conditions with the addition of each processing/composition variable [7,27]. For example, the exhaustive study of 10 compositions at 10 temperatures would only necessitate 100 experiments. Upon additionally investigating 10 pertinent pressure values and the inclusion of a new constituent at 10 concentrations, there appear now 10^4 possible experiments. This is often prohibitive for the implementation of an exhaustive or "grid" search of the sample space, and although it is quite obvious that an Edisonian approach would be inapt, it is also true that a purely *automated* grid-search may not be appropriate given the time/money costs [27]. This "curse of dimensionality" problem is highly prevalent in materials science, especially in the fields of materials discovery and property optimization since there are a large number of potential constituents (considering binary, tertiary, and quaternary compounds consisting of

elements selected from across the periodic table), a large number of processing parameters (temperature, pressure, etc.), a large number of structural considerations (film thickness, growth technique, crystal structure, etc.), and a large number of relevant property metrics (optical, electrical, mechanical, etc.) that affect any target performance criterion [14,27]. When considering the complex web of process/structure/property/performance relationships, is there a way to “intelligently experiment” to reduce the necessary number of experiments to achieve a CO; and how should one do this?

This is not a new question, and the Design of Experiment (DoE) field has existed to address the large sample space problem for many decades [32]. This field is concerned with optimal experiment design when under resource or time constraints, and aims to identify causal variables. It may explore relationships between independent and dependent variables in a multifactorial fashion, alter experimental design based on collected results (sequential analysis), and use randomization - among other techniques - to achieve this [33].

In AE, active learning takes the place of DoE [32]. Specifically, the quantification of value for a potential experiment - as mentioned earlier - is done through what is known as an *acquisition function*. In the closed-loop AE process flow, an acquisition function is generated and updated after each successive data point is measured, and signifies the *value metric* that each new potential experiment “should” have (*Section 1.3.2 for ML background*). The next experimental conditions can be chosen on the basis of the optimal value metric [27]. An acquisition function is meant to adapt and change after each successive data point is collected, reflecting a new “mindset” from which the robot works with as it incorporates new data. This is roughly akin to a human taking *interest* in guiding the experiment a certain or different way once they have conducted some runs/trials - something that purely automated systems are incapable

of achieving. The degree, however, to which this characteristic truly embodies the ineffable process of human *intuition* in guiding experiment is a subject of much debate [5]. Nonetheless, the acquisition function is the next logical step towards *approximating* the process of our adaptable intelligence and intuition. Since, inherently, the acquisition function is a *supposition* of experimental value it is of great importance that acquisition functions be modular, flexible, and to whatever degree possible, *informed* [16,34]. Without these characteristics, it is easy for the acquisition function to be ill-suited for providing a real benefit towards intelligently searching the space, and in fact certain comparative studies have shown little to no benefit at all from relatively uninformed active learning schema in guiding certain experiments [9]. To inform an acquisition function means providing it with prior data, physics axioms and principles (as in physics-informed AE), or even - in the case of human-in-the-loop AE - some human supervision to ensure that it doesn't become too *unreasonable* or counterproductive in its experimental suggestions [34,35]. Reasonableness, however, is quite hard to quantify, and to a large degree the predominant intention for implementing active learning/AE guidance in certain fields is that humans themselves may be ill-suited toward - and sometimes biased against - seeing the data-driven "signs" that point towards valuable next-experiments [27,36]. In summary, active learning, like any tool, has its pros and cons, and it should be the goal of any teaching program to develop a critical thinking mindset for distinguishing between its pros and cons so that it is properly applied in a useful manner (*Section 2.1 for Design Principles*).

1.3.2 Relevant ML Concepts

The ability to autonomously guide experimental design, as discussed in *Section 1.3.1*, relies on a quantification of supposed value as a function of experimental prerequisites, which in turn emerges from the underlying ML framework utilized. In this thesis, the predominant ML

frameworks used are that of Bayesian statistics and Gaussian processes. In our application of ML to active learning for AE, the task will often be of regression: minimization of the discrepancy between our models predictions (i.e., our beliefs) and the observed experimental data (i.e., reality). Our predictive models in the case of parameterized models may rely on the selection and tuning of a functional form, or in other cases (such as with Gaussian Processes) lack the need for an explicit functional form but instead require the selection of an appropriate kernel, which is essentially a covariance function [37].

The utility of Bayesian statistics with regard to active learning arises from its ability to quantify model *uncertainty* [37,38]. As discussed in *Section 1.3.1*, some COs (exploration, exploitation), explicitly require some encoding of the concept of what is known or unknown (i.e. where we are confident in our beliefs, and where we are not). These concepts can be conflated with the quantification of uncertainty in our model [37,38]. The Bayesian statistical approach is based on the work of Thomas Bayes, and is exemplified by Bayes theorem (*Equation 1.1*) [38].

$$P(A|B) = \frac{P(B|A)*P(A)}{P(B)} \quad (\text{Eq. 1.1})[38]$$

This equation is intended to assist in producing the *posterior probability* [$P(A|B)$], a conditional probability that represents the probability of event A given event B is true [38]. We rely on inputs of a *prior probability* [$P(A)$], which represents the initial beliefs about the probability of event A, and a *marginal probability*, [$P(B)$], which may be expanded into constituent terms to allow for calculation [38]. The Bayesian framework is used often in the analysis of positivity rates in medical testing, but to see its application to active learning, it may be rewritten (*Equation 1.2*) [38]:

$$P(model|data) = \frac{P(data|model)*P(model)}{P(data)} \quad (\text{Eq. 1.2})[38]$$

In this case we can presuppose a predictive model, and our *posterior* represents the probability of our predictive model being true given our observed data. If we recognize that our predictive model is a continuous explanatory function across our sample space, we can see that we actually calculate a *posterior distribution* across this same space [37,38]. This may be a corollary to a *confidence* distribution for our model (i.e. where we are more and less certain about its predictive capabilities and whether or not it is “true”). In addition to inputting a predictive model, we must also input a *prior* (which represents our initial beliefs/confidence for our model) [38]. Our other conditional probability [$P(\text{data}|\text{model})$] (which represents the likelihood of observing this data, assuming our model is correct) is the main metric that is updated as we gain new data. In this way, Bayes Theorem can be used iteratively between closed-loop runs (synthesis & measurement) to constantly update our *prior* and use the calculated *posterior* to construct our acquisition function (based on quantification of uncertainty) (*Figure 1.2*).

New data alters our prior beliefs → posterior beliefs

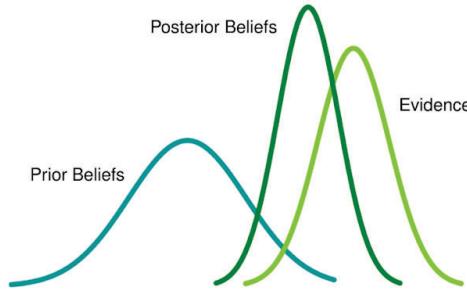


Figure 1.2: Pictorial representation of Bayes Theorem for iteratively updating our beliefs

Two of the active learning techniques implemented on LEGOLAS to study the simple chemical system described here (*Chapter 3*) involve Bayesian ML to some degree: Bayesian optimization using Gaussian processes (*Section 4.1*) and Bayesian Inference for parameter refinement in preselected models (not discussed in this thesis). The third technique - Hypothesis

Generation and Validation - utilizes an informational entropy based approach for quantification of experimental value (i.e. the acquisition function) (*Section 4.2*). To produce potential hypotheses, several candidates are either preselected by the user or generated via a symbolic regression package (genetic programming) (*Section 4.2.1*). Fitting of these candidate hypotheses occurs either through simple non-linear least squares regression, or automatically within the symbolic regression functions. Hypothesis validation is accomplished by evaluating one of two possible metrics, both of which promote fitness and penalize complexity: Bayesian Information Criteria (BIC), and M.S.E./Complexity score (*Section 4.2.1*). The fundamental statistical workings of symbolic regression will not be described in this paper, but more can be learned via sources in the bibliography [25,39]. Only the Bayesian Optimization active learning schema is currently taught in the *Machine Learning for Materials Science* course (ENMA 437/637) at the University of Maryland, where LEGOLAS is used as a teaching tool, although other techniques may be introduced in future implementations (*Chapter 3*).

1.3.3 Autonomous Experimentation in Materials Science

Many thorough reviews regarding the successes, experimental extents, and scopes of AE in theoretical, computational, and experimental materials science have been published in the past decade [7-9,12-14,16,18,27,29]. The use of AE for exploration, optimization, and discovery of materials systems has been already proven in many areas of materials science: reducing the number of neutron diffraction experiments for determination of transition temperatures of magnetic materials [28], incorporation of physics intuition and structural phase mapping to accelerate the discovery of best-in-class phase change memory materials [34], autonomous study of processing parameters to optimize carbon nanotube growth rate [40,41], autonomous optimization of mechanical properties of 3D printed structures [42], accelerated discovery of

new metallic glasses [43], active learning for tuning of interatomic potentials in atomistic simulations [44], among many others [29,45-49]

At the heart of it, the greatest misconception surrounding AE is that it could elicit an elimination of humans from the scientific process altogether [12,14,27]. In all these studies mentioned, however, a great deal of human guiding, monitoring, and encapsulation of human intuition into ML frameworks was required to create meaningful results [29,34,40-49]. The hopes of those developing AE systems are that they could (1) free up time for experts by alleviating them of laborious and repetitive laboratory procedures, (2) equip them with data-driven methods of navigating within highly dimensional sample spaces perhaps beyond the scope of human intuition & theory based analysis, and (3) forgo analytical delays *between* experiments to allow accelerated exploration with automated equipments.

The field of AE itself has its beginnings in Biology and Chemistry, where workflows generally involve more liquid handling and are less complex [17,50-52]. In the field of Materials Science, workflows may require high amounts of dexterity, and can be extremely complex and highly dissimilar from case to case [13,29]. Insights from thermodynamic phase stability, structural considerations, and defect equilibria (among other confounding factors) have proven necessary for making some sense of the inherently high-dimensional and highly-complex search spaces (process/structure/property/performance relationships) [9,34,35]. The call to utilize autonomous exploration techniques is therefore warranted, yet it is further complicated by expensive and complex characterization techniques [12-14]. Due to these restrictions, the use of Self-Driving Labs (SDLs) - controlling synthesis, measurement, and analysis functionalities - has been limited to universities and groups with large funding sources, resulting in highly specialized and extremely expensive setups that are not easily accessible to the majority of research groups,

and completely out of reach for most educational instruction purposes [49,53,54]. A few commercial modular SDL platforms are on the market, but again many of them cost hundreds of thousands of dollars, leading to a lack of availability for educational institutions [53,54]

In order to fully embrace SDL technologies, a multitude of expertise over many domains is needed including computer science, decision-making theory, experimental science, machine learning, statistics, materials science, systems engineering, robotics, software design, etc [19,55]. An ideal research group working in SDL technologies might consist of experts in various fields, but it is quite unreasonable to expect any one person to possess the utmost expertise in all necessary domains. An SDL compatible workforce should at least be conversant in these fields and understand the basics of the various fields [19,55] (*Section 1.3.4*). For developing a manufacturing workforce in industry to manage autonomously operating factories, the same is true [19,20]. Additionally, since collaboration is required to achieve the large-scale goals of autonomous manufacturing or experimentation, teamwork should be emphasized in developing the workforce [19]. One of the main barriers to the expansion of AE as a field, however, is the lack of availability of lower-cost and available teaching platforms and curricula for the education of an SDL compatible workforce [12,13,18,19,27,55] (*Section 1.3.4*).

1.3.4 Educating an SDL Compatible Workforce

The Materials Genome Initiative (MGI) is a large-scale multi-agency federal initiative launched in 2011, with the goal of “deploying advanced materials twice as fast and at a fraction of the cost compared to traditional methods” [55]. The MGI was developed in response to the 10-20 year delay time from materials discovery to commercial implementation that handicaps our ability to respond to existential societal threats such as climate change and sustainable energy needs [7,55]. In 2021, the MGIs strategic plan consisted of three main goals, one of which was

to “Educate, Train, and Connect the Materials Research & Development Workforce” [19,55]. From its conception, MGI set out to ensure our workforce was “trained for careers in academia or industry, including high-tech manufacturing jobs”[55]. The first objective to tackle in achieving this goal was to “Address Current Challenges in R&D Education” at the undergraduate, graduate, and foundational (K-12) levels. The second objective was to “Train the Next-Generation Workforce,” which includes mid and late career post-graduates, manufacturing workers, and other industry employees [20,55]. In 2019, an MGI working group identified 3 foundational pillars and their associated necessary competencies: data management, computation, and experimentation (*Figure 1.3*) [19,55].

Foundational Pillars	Key Competency Areas
 Data	<ul style="list-style-type: none"> • Data handling • Modeling and simulation visualization • Software and codes to manage MG workflows
 Computation	<ul style="list-style-type: none"> • Quantum and atomistic modeling methods • Microstructure evolution and material response • Multiscale and continuum modeling methods • Integrated workflows for computational tools
 Experiments	<ul style="list-style-type: none"> • Multi-objective design and decision-making under uncertainty • Measurement methods and tools • Sensor fusion, high-throughput methods, and automation

Figure 1.3: Three target foundational pillars for educational development identified by an MGI working group[19]

The MGI working group emphasized that “students need not be experts in all three areas but should be conversant in multiple topics across this spectrum” [55]. They also pointed out that the contrasting vernacular and cultures of theorists, experimentalists, computationally scientists, and other contributing experts necessitated a convergent educational preparation in order to provide the optimal cross-pollination between groups. Teamwork - a critical aspect of large scale success - was emphasized as well [55].

Existing educational solutions for the development of MGI-related skills have touched upon perhaps 1-2 of the foundational pillars, but rarely all three at once. For instance, at the graduate level, the Data-Enabled Discovery and Design of Energy Materials (D³EM) at Texas A&M University provides a wonderful preset track for the convergence of data informatics, computational methods, teamwork, leadership, and materials science fundamentals [56]. However, it currently lacks emphasis on automation and autonomous technologies: a crucial aspect of experimental acceleration techniques [56]. Although other established programs at the undergraduate and graduate level - as well as informational bootcamps and summer programs - have also begun to incorporate ML and active learning techniques into their curricula, they currently lack an available *tangible platform* from which to *experimentally* teach these techniques [56-61]. For the *Machine Learning for Materials Science* graduate/undergraduate course (ENMA 437/637) taught at the University of Maryland, this has been a challenge since the inception of the course in 2019. The majority of the course modules focus on ML and applied ML techniques, and at the end, when active learning is discussed, it was observed that many students lacked the ability to fully incorporate active learning as a decision making engine (either in the class assignments or into their own applied research).

In other developing fields such as drone technology and self-driving cars, where there is also a large emphasis on autonomous development, they have succeeded educationally by implementing tangible platforms in the classroom such as self-driving RC cars with multitudes of sensors and autonomously flying drones [62-66]. It was observed that the use of real platforms helped identify and remedy deficiencies in understanding of the robot point-of-view (POV) (i.e. what the robot knows vs. what the human knows), robotics hardware proficiency, coding and data management skills, and teamwork capabilities [66]. In developing these skills,

students became more adept at distinguishing between realistic and useful applications of autonomous control, and those that were “impractically ambitious”[66]. In this way, the students came to fundamentally deconvolute the seemingly nebulous concepts of AI, ML, and autonomous control to facilitate practical applications [66]. Although the use of simulated experiments or previously collected data can be helpful in teaching the data-driven side of autonomous systems, it can also lead to a lack of proficiency and understanding from the experimental side. It is a goal of MGI to have students be at least conversant in *all* of these fields [19,55]

Some of the tangible and hands-on platforms for autonomous drone technology were only a few hundred dollars and composed of widely available DIY robotics kits (Raspberry Pi, Arduino, etc.) [64]. However, in the case of autonomous driving, the F1Tenth program noted that their price per system (nearly \$5,000) was cost prohibitive (considering a class typically needed 10 systems) for university programs, and certainly at the foundational (K-12) levels [62]. Therefore, for maximum educational impact, it should be a goal of any tangible platform to be low-cost and affordable (*Section 2.1: Design Principles - Affordability*). At the same time, one of the main benefits of a hands-on educational tool is ultimately the captivation of student interest. It should also be a goal of the system to entice students and motivate them with exciting and fun challenges (*Section 2.1: Design Principles - Usability & Modularity*) [13]. In the case of the F1Tenth car, this was achieved by replacing exams with racing events [62]. Lastly, one key aspect of all tangible platforms is that they attempt to accurately portray “real-world” autonomous situations (*Section 2.1: Design Principles - Transferability*). For hands-on educational platforms, there must be a careful balance between system complexity and usability, so that the majority of students are not overwhelmed nor bored during its use [66].

Although SDLs are being developed at research institutions for fruitful materials science studies (*Section 1.3*), these bespoke systems are typically hundreds of thousands of dollars, so that they are neither available nor abundant enough for widespread educational use [13,49,53,54]. Several groups in the past decade have seen an opportunity to produce tangible SDL platforms for teaching at a wide range of costs (\$100 - \$10,000) (*Table 1.1, Figure 1.4*) [67-73]. Typically, these are cartesian style robots that use syringe pumps or peristaltic pumps to manipulate liquids, and perform characterization through contact measurements or via computer vision techniques (composition-color relationships, composition-acidity relationships, and even autonomously refining cocktails based on human feedback) [67,69-71,73-79]. There are also examples of autonomous 3D print optimization schemes, networking robots, and some attempts at solid-state experiments (using powders or wax) [68]. *Table 1.1* shows the strengths and drawbacks of each of these systems. Some SDL platforms are high quality, modular, usable, and transferable, but simply too expensive for educational use [67,72,76,77,79]. On the other hand, some are extremely cheap, but lack modularity, appeal, and most importantly *transferability* when assessed along the MGI emphasized competencies [19,55,68-70].

Table 1.1: Summary of some recent low-cost SDL platforms.

System (Year)	Size & Type	Cost	Application	Usability & Interest	Complexity of Construction and Use	Modularity
OpenTrons (2015) [67]	3'x3'x3' <i>Cartesian Style Robot</i>	~\$9,000 for unit, thousands more for accessory modules	High quality Liquid Handling and Pipetting Robot, has been used in MIT Machine Learning courses [71]	Highly usable, interesting to work with	Commercially constructed, Easy to use	High (however there is a high cost for modular parts)
Chemorobotic Robot (2014) [79]	3' x 2' x 8" <i>Cartesian Style Robot fed by Syringe pumps</i>	Estimated >\$5,000 due to use of expensive syringe pumps	Research on "evolution" of droplets through an autonomously guided generational study [79]	Moderately usable, potentially very interesting study for students	Built in research environment (high complexity), relatively easy to control with Python interface	Moderate (3D printed parts are utilized)
Bayesian Optimization Bartender (BOB) (2020) [70] & Networking Reaction Robot (2018) [69]	2' x 1' x 1' <i>Liquid Mixing stationary robot that uses peristaltic pumps.</i>	\$200-\$500, based on publicly available drink mixing robot [73] and \$500 from bespoke design [48] Price will vary with the number of sensors implemented and Raspberry-Pi/Arduino prices	Used to autonomously research different relationships involving making pleasant drinks (human input), learning pH responses, density space, and color response.	Highly usable, also very interesting for both students and adults.	Complex construction (requires soldering, wiring, use of peristaltic pumps). relatively easy to control with Python and ChemOS interface	Moderate modularity, mainly provided by adding additional sensors and heat beds. No cartesian movement available.
Additive Manufacturing Autonomous Research System (2021) [72]	2'x2'x2' Depending on size of printer. <i>3D printer with camera.</i>	>\$1000, depending on printer model used.	Used to autonomously optimize print parameters using computer vision.	Moderately usable, moderately interesting although not very hands-on	Complex to modify existing printer, use is moderately easy with	High modularity in terms of applicable ML algorithms, but not for physical system adaptation
SDL Demo (2022) [68]	3"x2"x2" <i>Micro-controller with LED and diode.</i>	\$100	Used to autonomously tune LED wavelength to target/desired output.	Highly usable, not very interesting or interactive.	Very easy to construct and very easy to use.	Very low modularity, single-experiment type setup

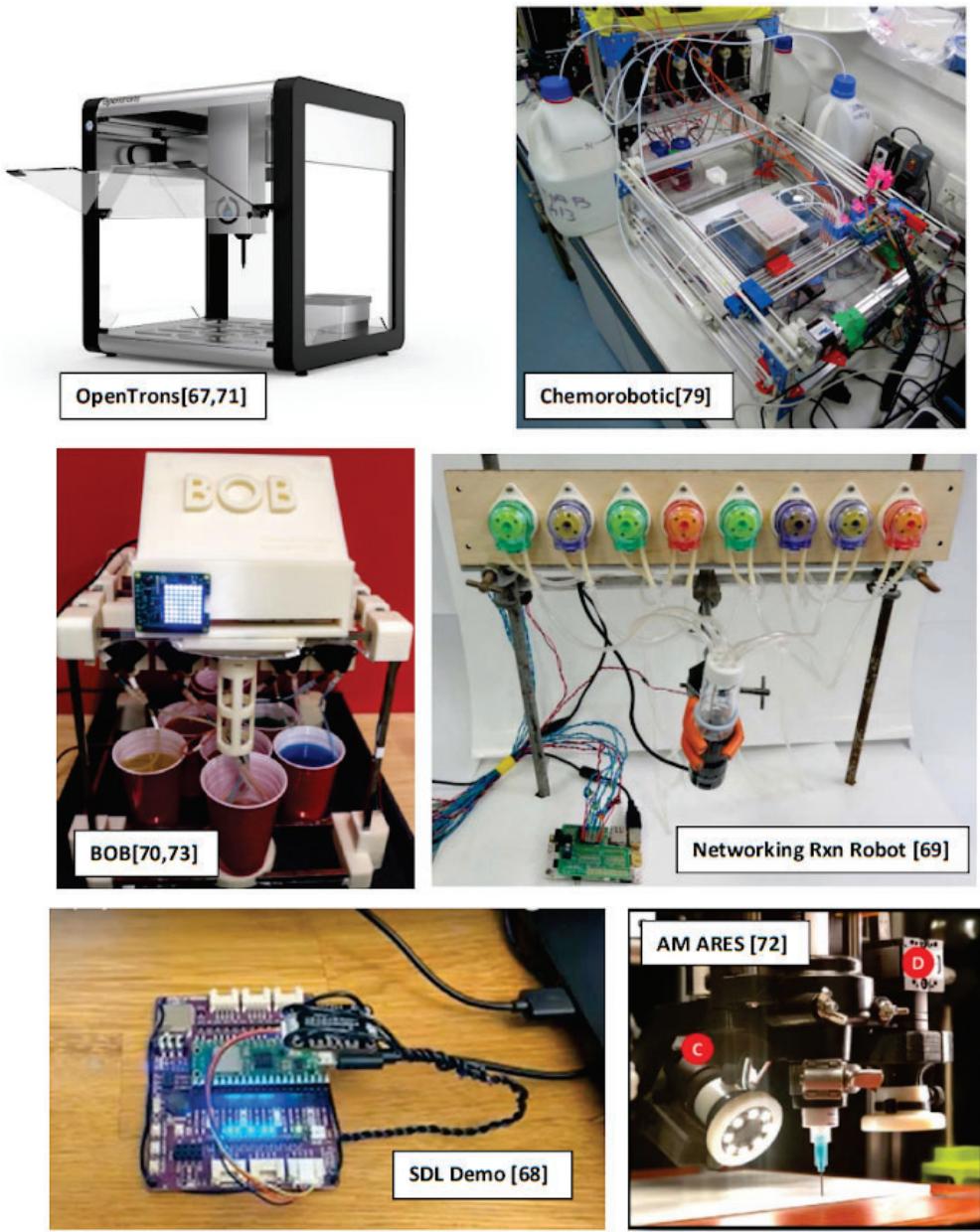


Figure 1.4: Images of low-cost SDLs from *Table 1.1*

In addition to the SDLs shown here, there has also been a rise in so-called “cloud labs” which allow students or collaborators to interact with experimental hardwares from a distance, in some cases using AI to facilitate complex manufacturing processes [54,80,81]. Although these technologies have the potential to revolutionize certain repeatable synthesis routes and improve upon the “crisis of reproducibility” increasingly apparent in recent human produced studies, they

are likely not good tools for educational purposes because they distance the student from the tangible aspects of experimentation [12,29,30,31]. In addition, it is clear from reviews and analyses of many SDL based setups and experiments that it is highly unlikely that humans will be completely removed from the physical laboratory[9,12,27,30]. In nearly every study it can be seen that a human is eventually needed - to replace experimental consumables, to troubleshoot hardware issues, etc. - and at best the “closed-loop” can only operate for so long before it needs tuning [13,15,17,29,46,51]. Additionally, thorough knowledge of experimental hardwares contributes to the reduction in “impractically ambitious” expectations often inherent to pure theorists [62]. Therefore, it should be a definite goal of educational platforms to continue to instill the importance of working with your hands and with tools on the equipment (*Section 2.1: Transferability*).

Of the best SDL platforms, (*Table 1.1*) the Chemorobotic Robot - essentially a liquid-handling cartesian gantry - was identified as the optimal option due to the incorporation of 3D printed modular parts, a captivating application, smaller profile, and robust design [79]. However, due to the incorporation of syringe pumps and bespoke parts, this SDL had an elevated cost - although cheaper syringe pump alternatives now exist [82] - and required a complicated construction process [79]. For the work represented in this thesis, a LEGO based chemistry robot design for STEM education (*Figure 1.1*) was developed to facilitate much of the same functionality at a reduced cost (~\$950: *For a full breakdown of parts and costs see Section 2.3*). This was done through 3D printing many of the parts, as well as replacing the syringe pumps with a simple plastic syringe. Computer vision and contact measurements were made possible by incorporation of a low-cost (\$30) reliable pH sensor and a small USB Camera [83,84].

LEGOLAS is a cartesian style gantry robot that can be expanded beyond the tasks elucidated in this thesis due to its high degree of modularity (*Chapter 5*).

The design principles followed for creating LEGOLAS are described in detail in *Section 2.1*, but briefly, the incorporation of LEGO parts was intended to elevate student interest in the project, remove the stigma of complexity from the physical system (so non-experts could learn as well), and allow for low-cost modularity (i.e. creative students could shape the course of the robotics design quickly and cheaply). By simplifying the experimental setup and reducing its costs, LEGOLAS facilitates financially achievable adoption at K-12, undergraduate, graduate, and industrial education levels, and effectively prepares students for the following competencies:

MGI Foundational Pillar 1 (Data)

- *Data Handling*: Students manage upcoming measurements and manipulate data structures to facilitate ML analysis. (*LEGOLAS Challenge 1 - 3 in Chapter 3*)
- *Software and codes to manage MG workflows*: Students embrace workflows in the easy-to-learn yet complexity tolerant Python language, where they manage both robotic control functions and active learning analysis in live-run closed loop AE fashion. (*LEGOLAS Challenge 4 in Chapter 3*)

MGI Foundational Pillar 2 (Computation):

- *Microstructure Evolution and material response*: Students study the relationship between composition in acidity in the main challenges described here. In a future exercise being developed for LEGOLAS (*Chapter 5*), students may be able to use computer vision to study the solidification kinetics and resulting growth structures of precipitating salt crystals in solution.

MG1 Foundational Pillar 3 (Experiments)

- *Multi-objective design and decision-making under uncertainty:* Students learn to quantify uncertainty using Bayesian ML techniques, and guide multi-objective optimization studies involving color mixing, solidification kinetics, and acidity.
- *Measurement methods and tools:* Students work hands-on with an electrochemical probe for measuring pH, calibrate the sensor, and build an understanding of the noise inherent to it (*Chapter 3*). In addition, they must calibrate the synthesis syringe using a mg sensitive scale and calibrate the sample well locations.
- *Sensor fusion, high-throughput methods, and automation:* This is LEGOLAS strongest strength, combining synthesis, measurement, and active learning into an autonomous closed-loop AE cycle (*Section 3.1.2*) (*Figure 1.5*).

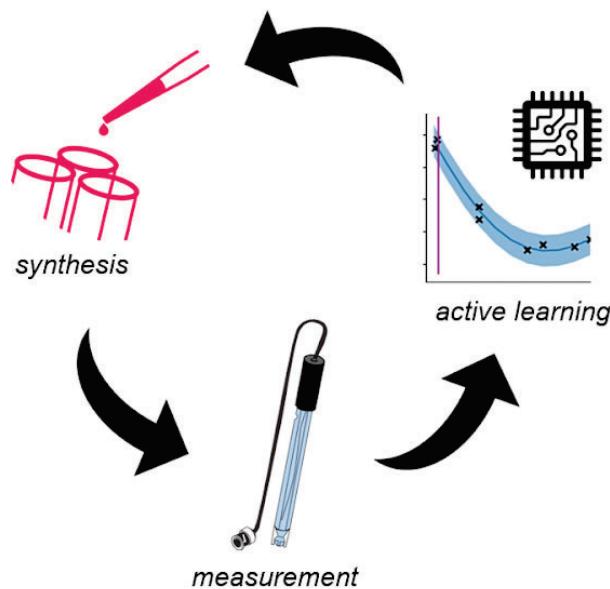


Figure 1.5: Simplified pictorial representation for a SDL such as LEGOLAS

1.4 Thesis Outline: Ch.2 - Ch.4 (Development and Operation)

An instructional overview of the construction and operation of LEGOLAS is outlined in *Sections 2.2 & 2.3*, with links to more detailed instructions therein. Worksheets and challenge templates for educational implementation are overviewed in *Chapter 3*. Code used to implement the active learning techniques for autonomous model discovery are outlined in *Chapter 4*. All detailed instructions files, worksheets, challenge templates, and code for any acquisition functions described here can be found centrally at the [LEGOLAS github page](#).

Inquiries for obtaining additional software modules for relevant chemistry experiments, challenge worksheets for students, active learning code, and machine learning code can also be requested via email from: takeuchi@umd.edu, agiladk@gmail.com.

Chapter 2: Systems Development

2.1 Design Principles

LEGOLAS is designed with several overarching principles in mind: affordability, usability, modularity, and transferability. These principles are held paramount for the purpose of achieving maximum educational impact. A large educational impact can best be realized through a system that is widely accessible and helpful in *realistically* portraying the benefits, drawbacks, simplicities, and challenges of closed-loop AE. It should also be a goal of the educational tool to promote critical thinking - with respect to active learning and experimental design - and not merely a prescriptive carrying out of established steps and rules (*Chapter 3*).

Most closed-loop AE systems are currently cost-prohibitive, leading edge research tools that are available only to a small, select few scientists and researchers who are far along in their careers [13,30]. A low-cost, easily reproducible AE system encourages the instruction of AE fundamentals at earlier ages; facilitating implementation in high-school, undergraduate, and graduate education. It is likely true that an early introduction - and thereby a wider berth - for the AE field could only serve to enliven and further enrich its goals of improving our ability to produce societally beneficial materials [7,13,30].

Usability of an educational AE system refers not only to its ability to operate reliably and smoothly, but also to its propensity for the captivation and inspiration of student interest [13,62,66]. LEGO components were selected to elevate this level of interest, as well as reduce the stigma of complexity that can often serve to overwhelm certain students. These LEGO components, however, are by no means of poor quality, and provide the robot with structure, modularity, and a pleasant aesthetic quality. All aspects of the design embody the principles of interactive usability and simplicity: manual controls on motors, open and accessible joints,

removable trolley & bridge, light-weight & sturdy aluminum frame, Arduino & Raspberry Pi integration, remote Wi-Fi connectivity, and Python based Jupyter Notebook control.

The modularity of an educational AE system is important to ensure it can be extended beyond its original mode of use into new and fresh domains for exploration by creative students [13,30]. Physically, the LEGOLAS system described here is quite modular: 3D printed (and thereby replaceable) components, semi-permanent joining methods (LEGO pins, machine screws), LEGO parts & motors that can be rearranged for different types of experiments, and an open frame and trolley design providing space for additional sensors, probes, cameras, and other experimental hardwares. These attributes allow LEGOLAS to be expanded into new experiments beyond the ones described here (*Chapter 5*). With respect to the software side of an AE education system, ideally there should be a control interface that is relatively easy to learn, robust, and tolerant of complexity [30]. For LEGOLAS, a Graphical User Interface (GUI) has been developed to reduce experimental calibration complexity. Coding in the easy to learn & widely supported Python language (Jupyter Notebooks) allows for both beginner and experienced programmers to get what they need out of the platform. Within Python, community supported active learning/ML and robot control functions can coexist, allowing students to visualize the process-flow of closed-loop AE quite well.

Finally, the aspect of transferability - albeit hard to quantify - is key to setting up an educational AE system for maximum impact. Transferability refers to the educational AE system's ability to accurately *approximate* the challenges, drawbacks, and benefits of a larger-scale research grade AE system in a way that will prepare students for the "real thing." The key fundamentals of research level closed-loop AE will be elaborated upon later (*Chapter 3*), but briefly, they are:

- (1) the skills of working with your hands & with tools on hardware,
- (2) the knowledge to interface with motors, sensors, & robotic components,
- (3) the patience and attention to detail to properly calibrate and tend-to the machine,
- (4) the knowledge of ML techniques for prediction, clustering, etc.
- (5) the skills of programming and data manipulation
- (6) the insight to encapsulate research “objectives” into active learning schema,
- (7) the foresight and knowledge to simulate live-runs,
- (8) the skills of presenting data in a digestible & meaningful fashion,
- (9) and the discernment to understand pros & cons of AE.

These are skills that are often distributed amongst a team of specialists in larger-scale research-grade AE groups, but all are necessary to ensure a successful, and *meaningful* use of AE for research purposes [19,66]. By possessing many or all of these skills, an individual has a greater ability to integrate their research goals and experimental processes to achieve relevant applications. However, nearly all high-level research occurs in group settings, so it should still be a goal of an educational AE system to promote teamwork [19]. In LEGOLAS, the principle of transferability is largely affected by the methods in which it is used/taught, rather than any physical characteristics of the machine itself. This will be discussed more in the educational section (*Chapter 3*). However, certain aspects of the LEGOLAS system - as with any machine - are liable to occasionally malfunction. These challenging moments happened to provide some of the greatest learning benefits for any of the students, since, in reality, no AE system is truly a “closed” loop (humans will need to periodically intervene) [13,30,46,51].

2.2 Mechanical Design

LEGOLAS is a cartesian gantry style robot with a frame track (y-axis), a bridge track (x-axis), and a trolley cart that rests on the bridge. Each assembly contains an assortment of LEGO components, aluminum frame extrusions, electronics, and/or 3D printed parts. Trolley, bridge, and frame may be constructed from the constituent components with common tools and adhesives (*Section 2.3*) and are easily removable for maintenance or alterations.

The X & Y position of the trolley cart can be controlled via keyboard or manually through control knobs (*Figure 2.1, 2.5*). Force sensors aligned along the X & Y axes allow for automatic recalibration of relative experimental coordinates (i.e. the location of the sample wells or reservoirs) in the case of the bridge or trolley being derailed or dislodged. The trolley cart contains all experimental components (pH Sensor, Syringe & Plunger, and/or camera), which are free to move in the Z-direction (*Figure 2.1, 2.2*).

The LEGOLAS was inspired by a fully LEGO based liquid-handling robot developed for chemistry experiments [85] (*Figure 2.11*). I have developed 4 generations of LEGOLAS since 2019, attempting to create each new generation at a lower cost, higher stability, and with greater interactivity possible. The chronological development and modifications made for each of these models is shown in *Section 2.5*. To better visualize the operation of LEGOLAS (*Generation III*), one can see it performing experiments in this [video](#) [86].

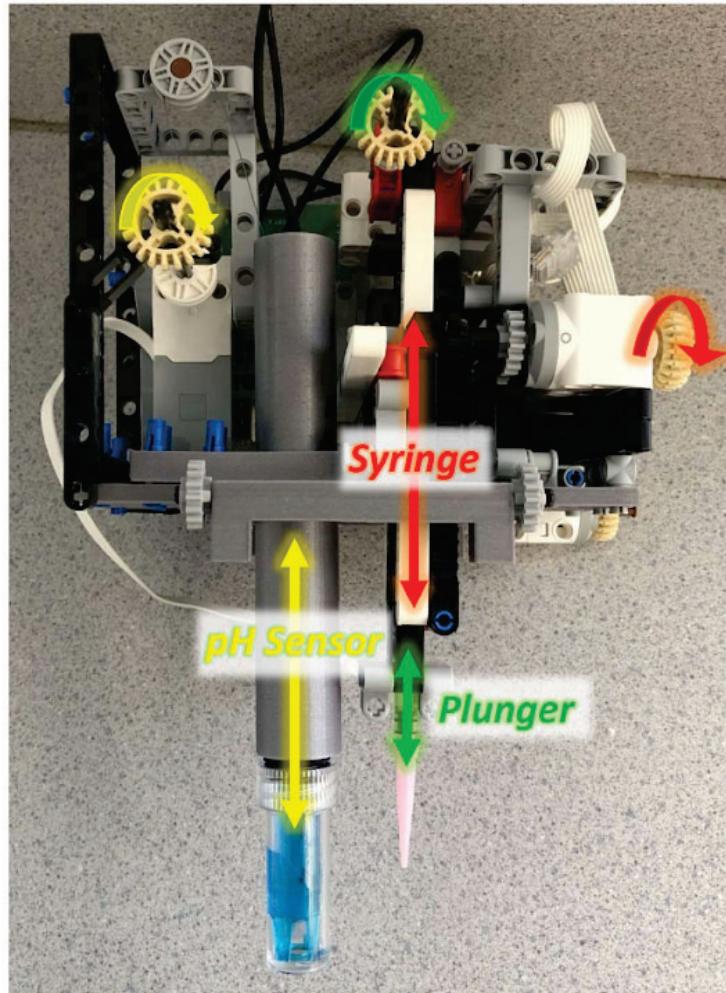


Figure 2.1: Experimental components on the trolley (*Generation IV LEGOLAS*). Axes of motion (*double sided arrows*) and manual control knob locations (*rotational arrows*) shown.

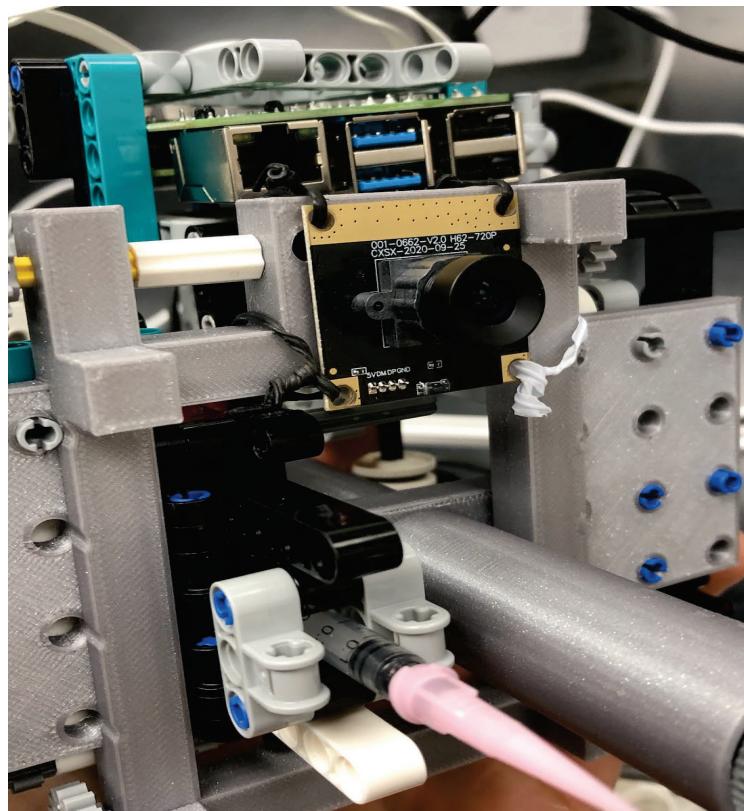


Figure 2.2: USB Camera attached to the bottom of the Trolley (*Generation IV LEGOLAS*).
The camera can easily be attached and removed for different types of experiments/studies.

2.3 System Construction & Preparation

This section contains the general process flow of sourcing components, 3D printing parts, constructing LEGOLAS, and preparing the experimental consumables needed for the simplistic chemistry experiment described in *Chapter 3*. Guidance to in-depth instruction modules are provided as links within each section.

Sourcing Components and Cost Breakdown:

The LEGO Components used in LEGOLAS design were originally sourced from the LEGO® MINDSTORMS® EV3 Core (45544) and Robot Inventor (51515) sets, but can be sourced more easily from individual part retailers and via the LEGO Education website (for motors and sensors). Aluminum frame components can be acquired from MakerBeam B.V., and additional non-LEGO electronics and chemistry equipment from Amazon and other online retailers. See the detailed links for specifics on part sourcing. A cost breakdown is shown in *Table 2.1*. Note that this does not include the cost of chemicals for experimentation or for a USB compatible camera, which can vary based on the application desired but typically lead to a full cost of around \$1,000.

Detailed Links: [LEGO Parts List](#), [Sourcing & Cost of LEGO Parts](#), [Sourcing & Cost of Non-LEGO Parts](#)

Table 2.1: Cost Breakdown for a single LEGOLAS robot

Category	Included	Cost
LEGO Parts	Structural Components	~ \$60.00
LEGO Electronics	Motors, Force Sensors	~ \$280.00
Frame Components	MakerBeam Structural Parts	~ \$100.00
Non-LEGO Electronics	Raspberry Pis, BuildHat, Arduino, pH Sensor, Chargers, Wiring	~ \$470.00
Chemistry Equipment	Wells, Stand, Syringe, Dispenser Tips	~ \$40.00
	Total:	\$950

Printing Parts: Parts were printed with Ø1.75 mm Galaxy Silver Prusament Polylactic Acid (PLA) using a Fused Deposition Modeling (FDM) style Original Prusa i3 MKS3 printer (*Figure 2.3*). With the exception of one Raspberry Pi Holder Case [87], all parts were developed in the Autodesk Fusion360 CAD software. Total PLA needed for all prints is ~410 g, for around \$12.50 in material costs. *Table 2.2* summarizes some relevant characteristics for each part. Information on optimal orientation, advanced printer settings, post-processing tips, as well as the CAD (.f3d) and printable (.stl, .3mf, .obj) files for each part are in the detailed links below.

Detailed Links: [3D Printing Guide](#), [Printable Files](#)

Table 2.2: Summary of FDM printed parts

Part	Purpose	Supports?	Quantity	Material Used (g)	Cost*
pH Sensor Guide Tube	Guide pH sensor towards sample wells	Yes	1	42.45	\$1.28
pH Sensor Sleeve	Protect pH Sensor	No	1	11.29	\$0.34
Trolley Base	Hold experimental hardware	No	1	38.71	\$1.17
Mid-Axle Supports	Support mid-spans of X-axis axles	No	6	14.77	\$0.45
End-Axle Supports	Support ends of X-axis axles	No	4	12.21	\$0.37
Side Assembly	Hold R-Pi & Force Sensors on Bridge	Yes	1	66.42	\$2.01
Reservoir Tank and Stands	Contain liquid constituents	Yes	1	203.41	\$6.15
Force Sensor Holders	Fasten force sensors to frame	No	2	2.18	\$0.07
R-Pi Case	Hold Raspberry Pi	No	1	16.51	\$0.50

*Based on \$30/kg PLA costs

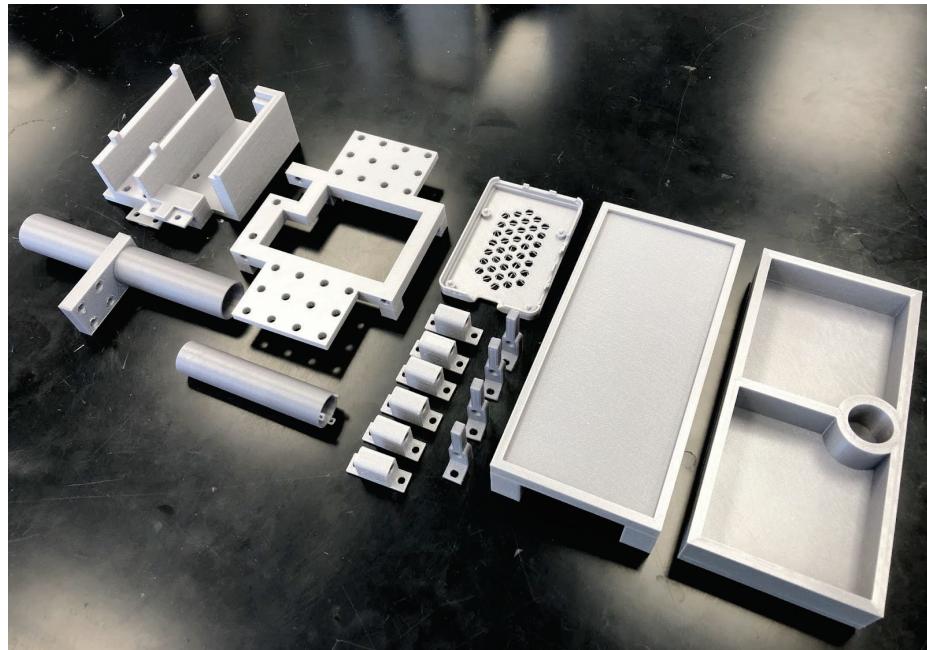


Figure 2.3: All 3D printed parts needed to build one LEGOLAS robot

Constructing Stand: The gantry stand is composed of MakerBeam (10mm x 10mm profile) miniature slotted aluminum extrusion pieces, 2 force sensor holders (3D printed), MakerBeam fasteners, a LEGO force sensor, and LEGO gear rack pieces (composing the y-axis of the gantry) (Figure 2.4). Store bought superglue and a 2 mm hex key are all that are needed to construct it. Instructions are included in the links below.

Detailed Links: [Stand Instructions](#)

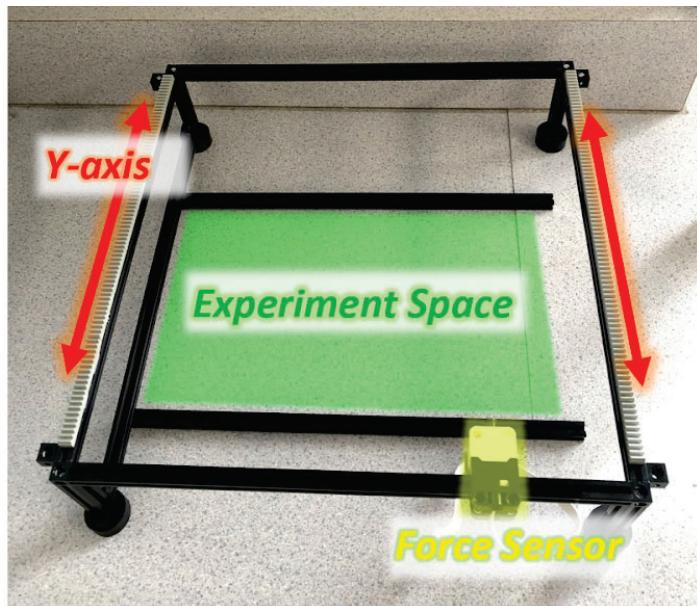


Figure 2.4: Fully assembled stand (320 x 340 x 60 mm profile)

Constructing Bridge: Similar to the frame, the bridge is constructed of an aluminum MakerBeam frame with associated fasteners (*Figure 2.5*). It also includes a LEGO force sensor, 3D printed components (4 End-Axle supports, 6 Mid-Axle Supports, 1 Side Assembly), assorted LEGO pieces, a Raspberry Pi, a BuildHAT, and LEGO rack pieces (composing the x-axis of the gantry). Like the stand, it only requires the hex key and superglue to be assembled.

Detailed Links: [Bridge Instructions](#)

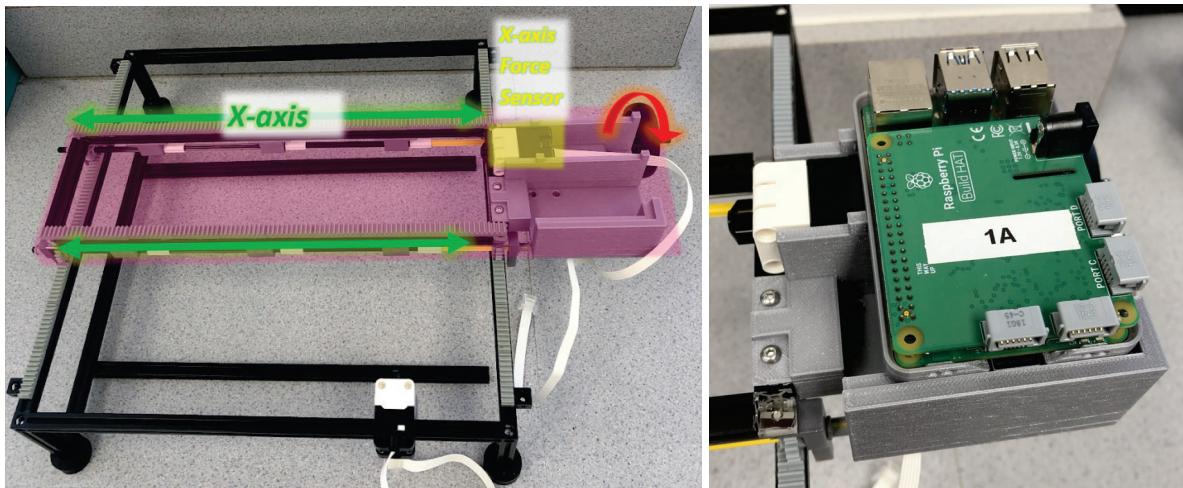


Figure 2.5: (left) Assembled bridge, with X-axis (double sided arrows) and Y-axis manual control gear location (rotational arrow) shown. (right) 3D printed Side Assembly with Raspberry Pi + Buildhat attached in their holder piece.

Constructing Trolley: The trolley is built of assorted LEGO pieces, 4 LEGO motors, a Raspberry Pi, a BuildHAT, and experimental equipment (plastic syringe, Arduino pH sensor, USB camera) that all rest on the 3D Printed Trolley Base frame (*Figure 2.6*). One must first build the Syringe Plunger Assembly, then the Syringe Holder Assembly, and finally follow the Trolley Instructions to complete construction. The only additional components needed to build it are scissors, superglue, 18-24" of braided fishing line, and 3 small barrel swivels.

Detailed Links: [Trolley Instructions](#), [Syringe Plunger Assembly Instructions](#), [Syringe Holder Assembly Instructions](#)

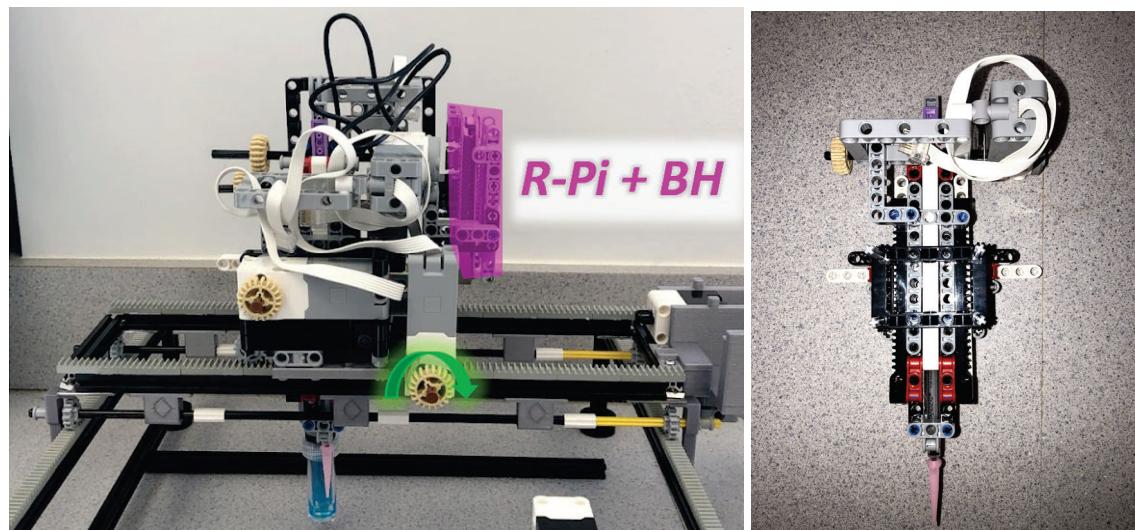


Figure 2.6: (left) Assembled trolley resting on the bridge, with Raspberry Pi + BuildHAT assembly and X-axis manual control gear location (*rotational arrow*) shown. (right) Syringe Plunger & Holder assembly, prior to being inserted into the Trolley Frame.

Sample Space: The sample space shown in *Figure 2.7* is applicable to the Henderson-Hasselbalch pH study (*Chapter 3*), and can be modified in terms of components and their orientation for other types of experiments. This setup contains the constituent reservoir tanks (3D printed and coated with an epoxy to ensure solution impermeability for acid, base, and Deionized (DI) water solutions), 48 sample wells (each Ø1.5 cm, 3.5 mL volume), and a mg accurate digital scale for volume calibration (*2.4: System Calibration Section*). Each reservoir tank holds 50 mL of either acid or base solution, facilitating the experimental use of all 48 sample wells if needed (assuming 2 mL sample volume). These components are held in place by two adjustable MakerBeam aluminum extrusion pieces, and can easily be removed from the sample space by sliding them out through the open end for post-experiment clean-up. The sample wells and reservoir are elevated several cm off of the ground, providing ample space for the insertion of hot plates and other process-altering equipment for modular experiment design, if desired (*Chapter 5*).

Detailed Links: [Chemistry Guide](#) (step #2)

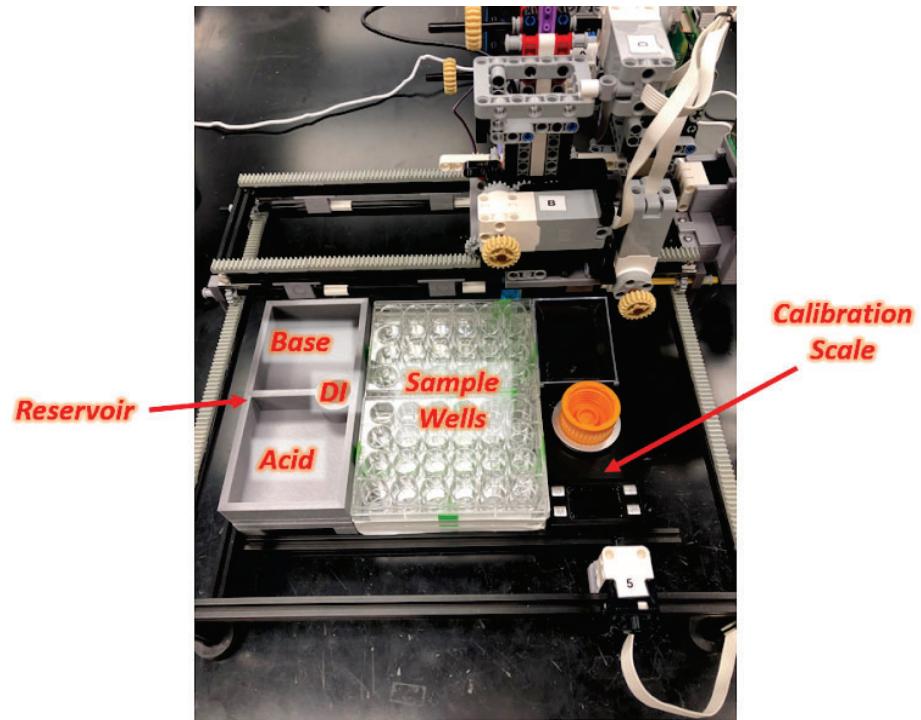


Figure 2.7: Sample space setup for the Henderson-Hasselbalch pH study

Chemistry: Preparing the solutions for the Henderson-Hasselbalch pH study involves non-toxic constituents, and produces chemicals with acidity on the level of vinegar and milk of magnesia, allowing safe classroom use even at primary school levels. Only a mg accurate scale, 10 mL graduated cylinder, small glassware components, and common chemicals are required (*see Chemistry Guide for more details*).

Detailed Links: [Chemistry Guide](#)

2.4 Connecting to & Calibrating LEGOLAS

This section describes the means of properly wiring LEGOLAS, downloading the appropriate configuration files, connecting to the robot via WiFi, and calibrating both the system and pH sensor (only needed if running acidity based studies as in *Chapter 3*).

Wiring: LEGOLAS has 2 Raspberry Pi & BuildHAT stacks, with one (*R-Pi A*) located on the Side Assembly of the bridge (controls Y-axis motor, Y-axis force sensor, X-axis force sensor), and one (*R-Pi B*) located on the Trolley (controls Syringe Z-motor, Syringe Plunger motor, pH Sensor Z-motor, X-axis motor). Two 120V wall plugs must be available for the BuildHAT chargers, which are the exclusive power source for LEGOLAS and the Arduino, which connects to the pH sensor and (*R-Pi A*). A chemistry lab retort stand is used to suspend the pH sensor's BNC cable and keep it out of LEGOLAS's path of motion (*Figure 2.8*).

Detailed Links: [Wiring Guide](#)

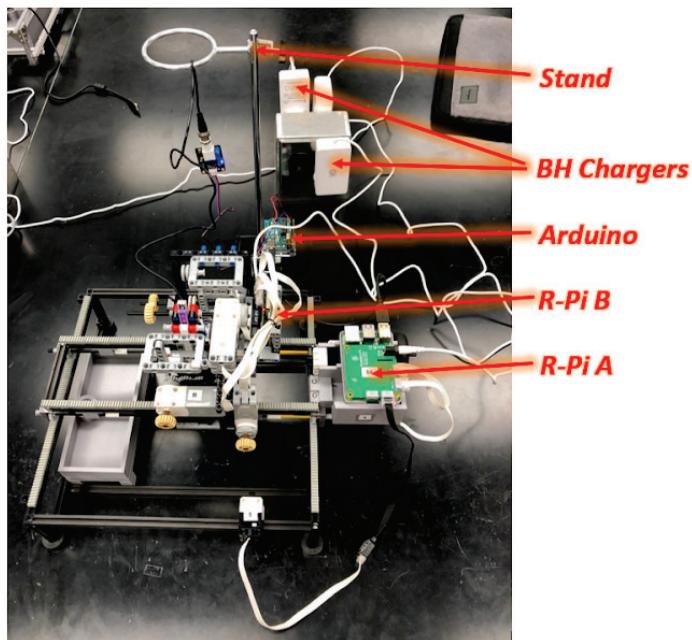


Figure 2.8: Location of relevant wires and electronic devices for LEGOLAS

Raspberry Pis & WiFi Connection: The Raspberry Pi's + BuildHAT stack must have the Raspberry Pi operating system installed, as well as packages for automatic remote SSH connection (RPyC Server) and BuildHAT interfacing. A WiFi connection (from your Computer to LEGOLAS) may be enabled through a local router to facilitate DHCP Client List observations and Address Reservations. Raspberry Pi configuration can also be copied from one microSD card to another using a USB microSD card reader, for expediting the process when setting up the second Raspberry Pi + BuildHAT stack (*see Raspberry Pi & BuildHAT Setup Guide for more details on proper setup*)

Detailed Links: [Raspberry Pi & BuildHAT Setup Guide](#)

File Set-up: The preferred LEGOLAS programming environment uses Jupyter Notebooks through the Anaconda platform. Files for calibration (manual.py, config.py), movement functions (core.py), demonstration (LegolasDemo.ipynb), and classroom use (LegolasOutline.ipynb) can be downloaded from the LEGOLAS github page (*see LEGOLAS Scripts*). Instructions for directory management and use are in the beginning of the LEGOLAS Calibration & Use Guide.

Detailed Links: [LEGOLAS Scripts, LEGOLAS Calibration & Use Guide \(pages 1-7\)](#)

System Calibration: Prior to using LEGOLAS for experiments, one must calibrate the location of relevant experimental locations (reservoirs, sample wells, DI water tank, syringe/pH sensor device offset) in cartesian space relative to a user set origin point (which is defined by an XY offset from the X-axis and Y-axis force sensors). If the LEGOLAS trolley or bridge is then derailed, it may reestablish these locations relative to that origin point. Once XY positions are

calibrated, the user must define the range of motion (in the Z-direction) for the pH sensor and syringe, as well as their distance of offset in the XY plane. Finally, the liquid-volume/gear-step ratio is defined via calibration with a mg accurate scale. One may use keyboard controls or the manual control gears to perform calibration steps. All aforementioned steps use a GUI that reduces calibration time to ~10 minutes (*Figure 2.9*). The calibration values are exported from the GUI into *config.yaml* (*to be called in the Jupyter Notebooks*). Additionally, the pH sensor should be calibrated (*see Arduino pH Calibration Code*) prior to experiment, which can be carried out with installation of the Arduino IDE and a 2-point buffer solution calibration. To extend pH sensor bulb life-time, it is advised to avoid long periods of exposure (> 1 hr) outside the pH storage solution (1 M KCl).

Detailed Links: [LEGOLAS Calibration & Use Guide](#), [Arduino pH Calibration Code](#)

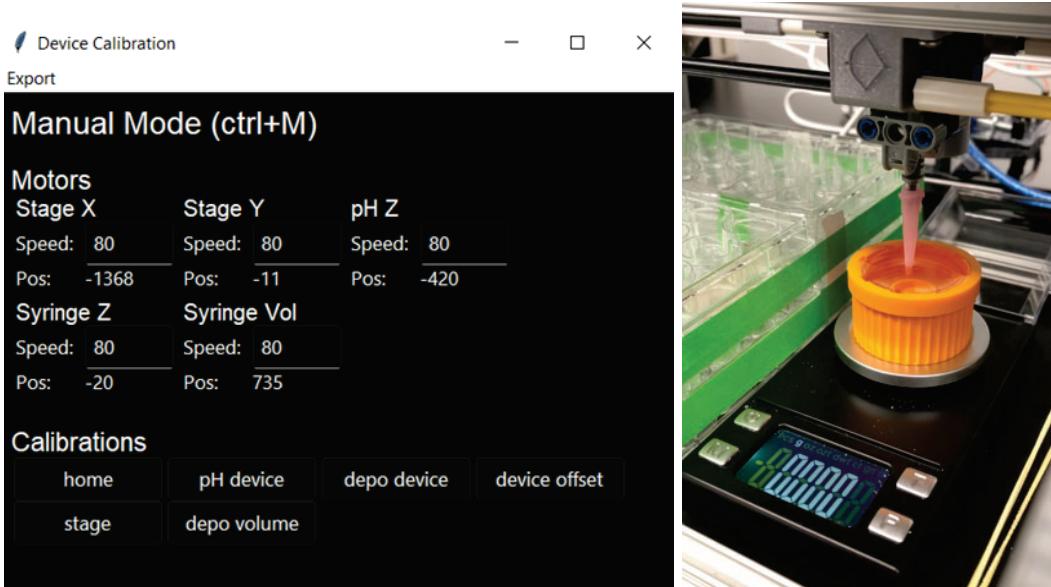


Figure 2.9: (left) GUI calibration window, and (right) liquid-volume/gear-step calibration using the mg accurate digital scale.

Coding Interface: Once the system is physically calibrated, one can connect to verify calibration of LEGOLAS using the *LegolasDemo.ipynb* file in Jupyter Notebooks (*Figure 2.10*). Students may also become comfortable with the motor movement functions here, before moving onto *LegolasOutline.ipynb* (which contains the worksheet problems in Markdown text) for completion of the assigned challenges. Installation and usage of common ML packages (such as GPy and scikit-learn) in this Python environment allows students to visualize the process flow of AE in one central location.

Detailed Links: [LegolasDemo.ipynb](#), [LegolasOutline.ipynb](#)

```
1 stage.move_to_cell(0,0)
2 pH_device.to_zpos("full_down")
3 pH_device.to_zpos("full_up")
```

```
1 for row in range(4):
2     for col in range(6):
3         depo_device.acquire(0.6, location="acid")
4         depo_device.deposition(0.6, row=row, col=col)
5         pH_device.move_to_cell(row=row, col=col)
6         pH = pH_device.pH_measure(stable_time=1)
7         print(f"row {row} col {col} pH {pH}")
```

Figure 2.10: (top) Example usage of fundamental movement functions (*contained in core.py*) and (bottom) example of a simple loop for experimental synthesis and measurement in a 4x6 grid of sample wells (no active learning involved). These snippets are included in the *LegolasDemo.ipynb* file.

2.5 Developmental Stages

This section visually depicts the 4 generations of LEGOLAS, in which mechanical design, interface methods, and reliability were iteratively improved based on student feedback and for the purpose of reducing material costs (*Figures 2.11- 2.17*)

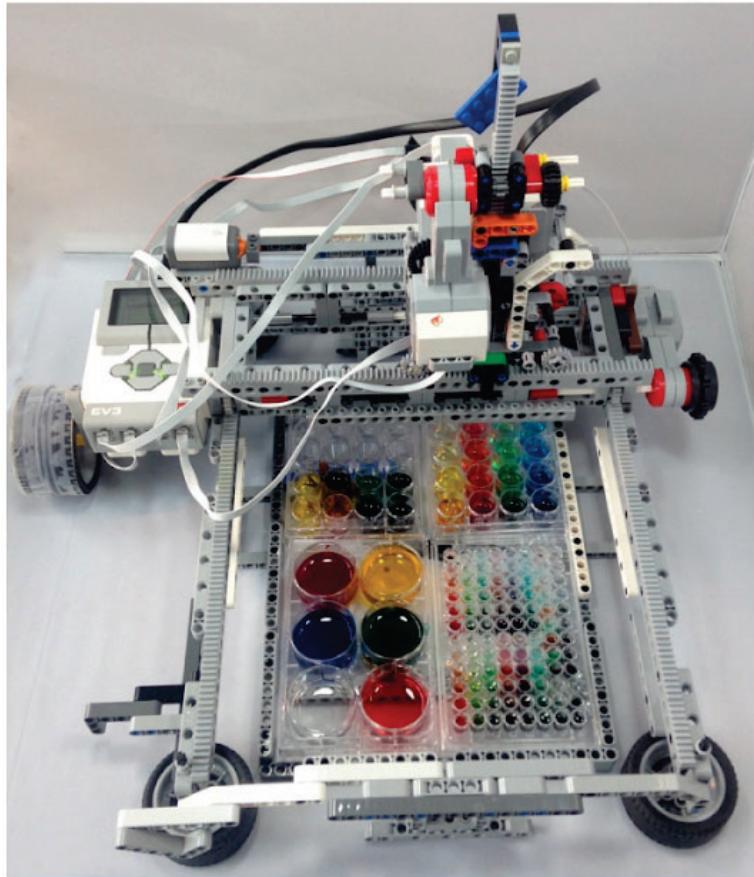


Figure 2.11: Liquid Handling robot that inspired the LEGOLAS design [85]. This Robot was designed using LEGO® MINDSTORM® EV3 robotic and frame components, and was capable of liquid dispensing and color detection (using LEGO color sensors).

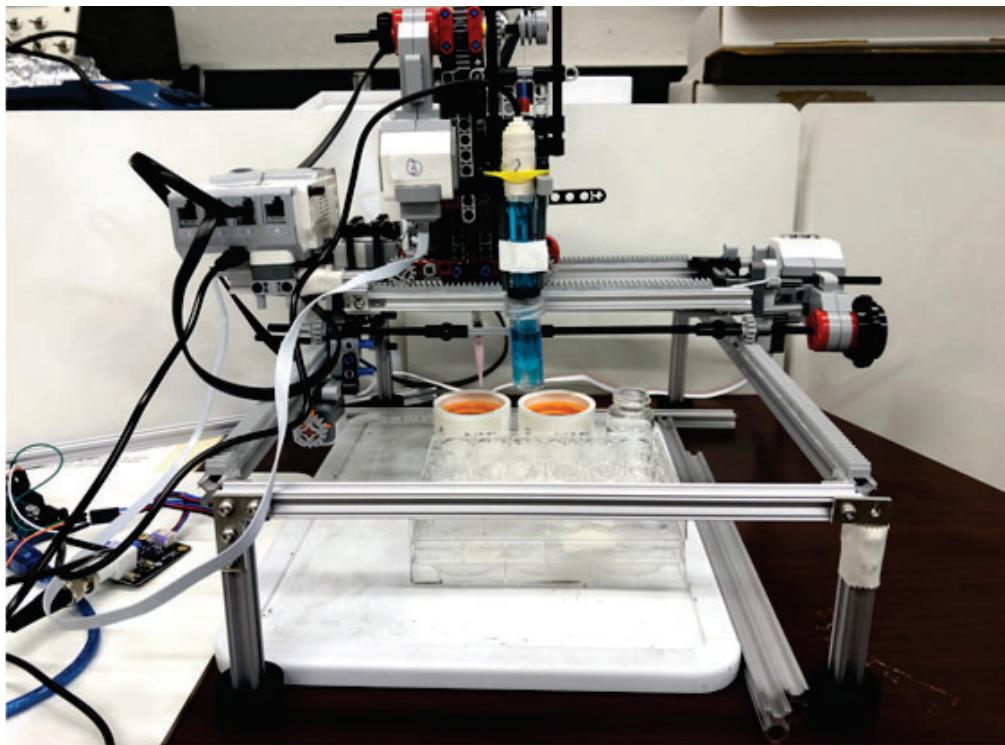


Figure 2.12: LEGOLAS (*Generation I*) using a Mindstorm EV3 Brick. This model was accessible for control via WiFi or Bluetooth, and now included a sturdier aluminum frame, as well as a pH sensor for acidity measurements.

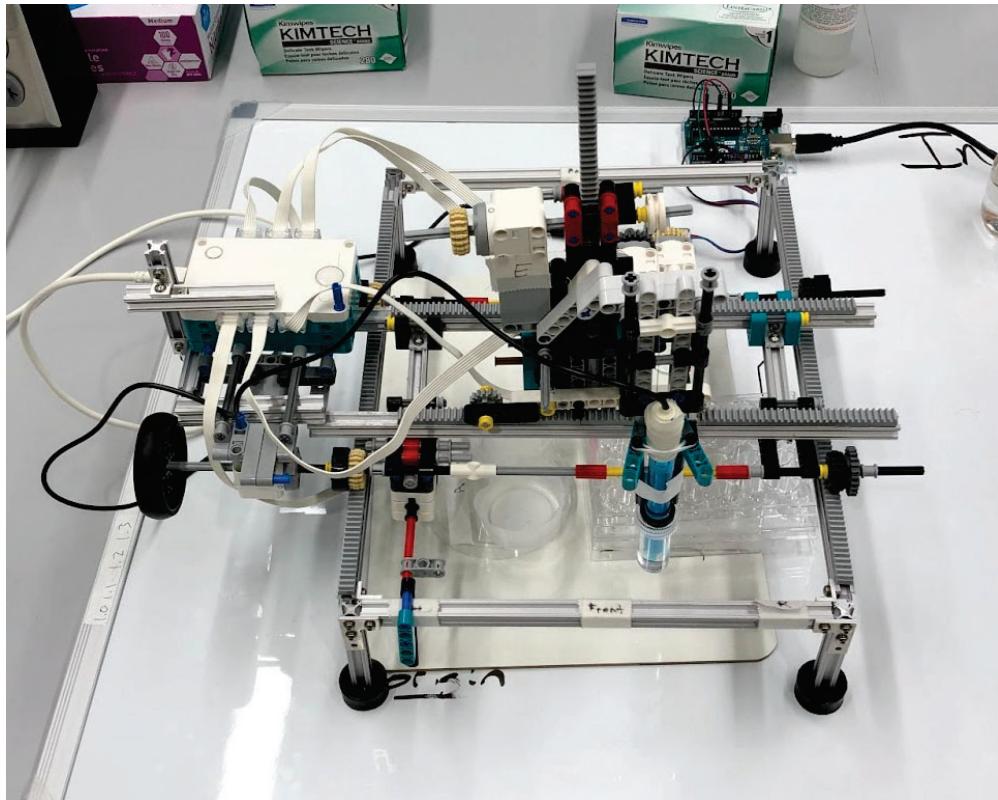


Figure 2.13: LEGOLAS (*Generation II*) using the LEGO® Robot Inventor Hub. This model was accessible for control via Bluetooth only, and was created to utilize LEGO's newest robotic kit as they phased out production of the MINDSTORM® EV3 sets. The Inventor hub acted only as a microcontroller, providing limited functionality compared with the EV3 kit.



Figure 2.14: Classroom Setup for Fall ‘21 Implementation, which used *Generation I & II* LEGOLAS.

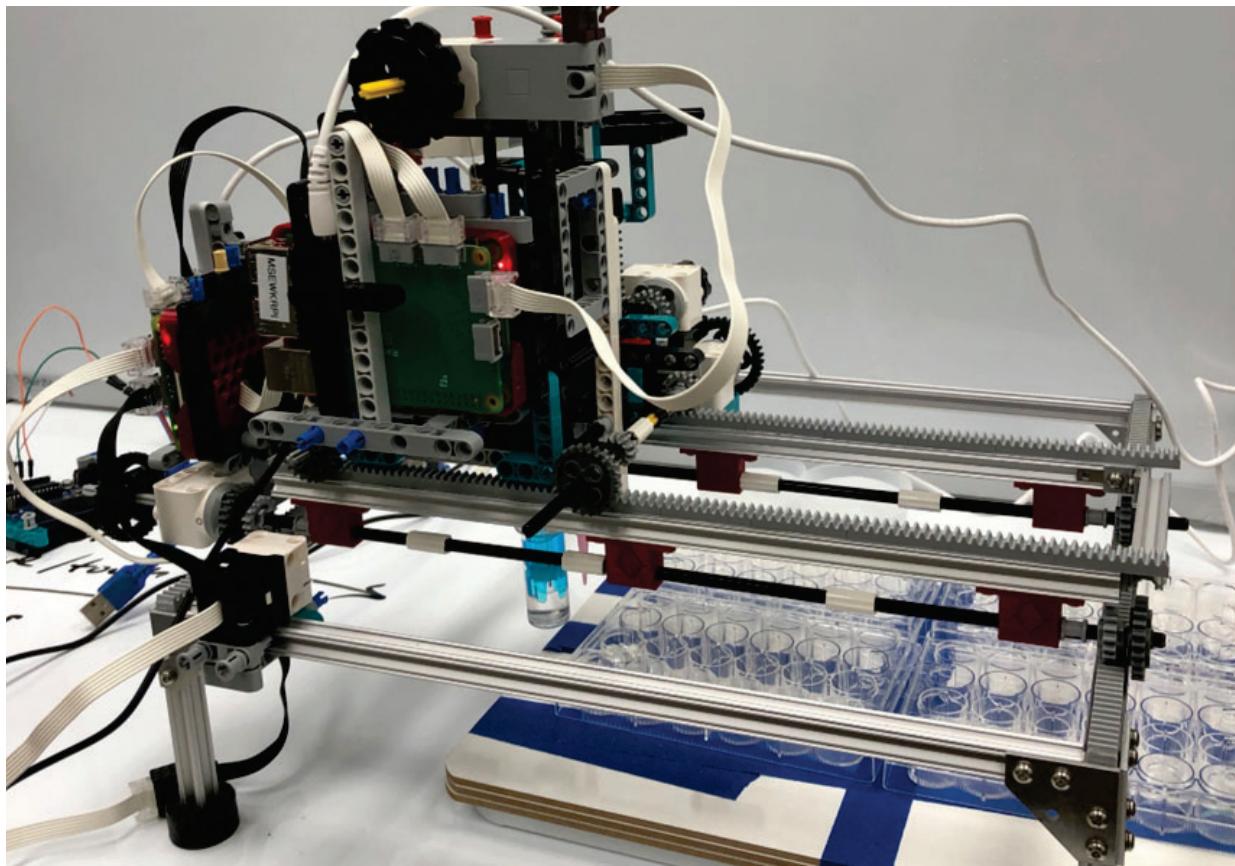


Figure 2.15: LEGOLAS (*Generation III*) using integrated Raspberry Pi & BuildHATs to interface with LEGO motors. This model was accessible for control through a SSH via WiFi. This design featured new 3D printed axle supports and greater mechanical stability. It was utilized to conduct the first live runs of the on-the-fly hypothesis validation active learning schema. The Raspberry Pi & Buildhat interface allowed the same programming freedom available with the prior MINDSTORM® EV3 kits.

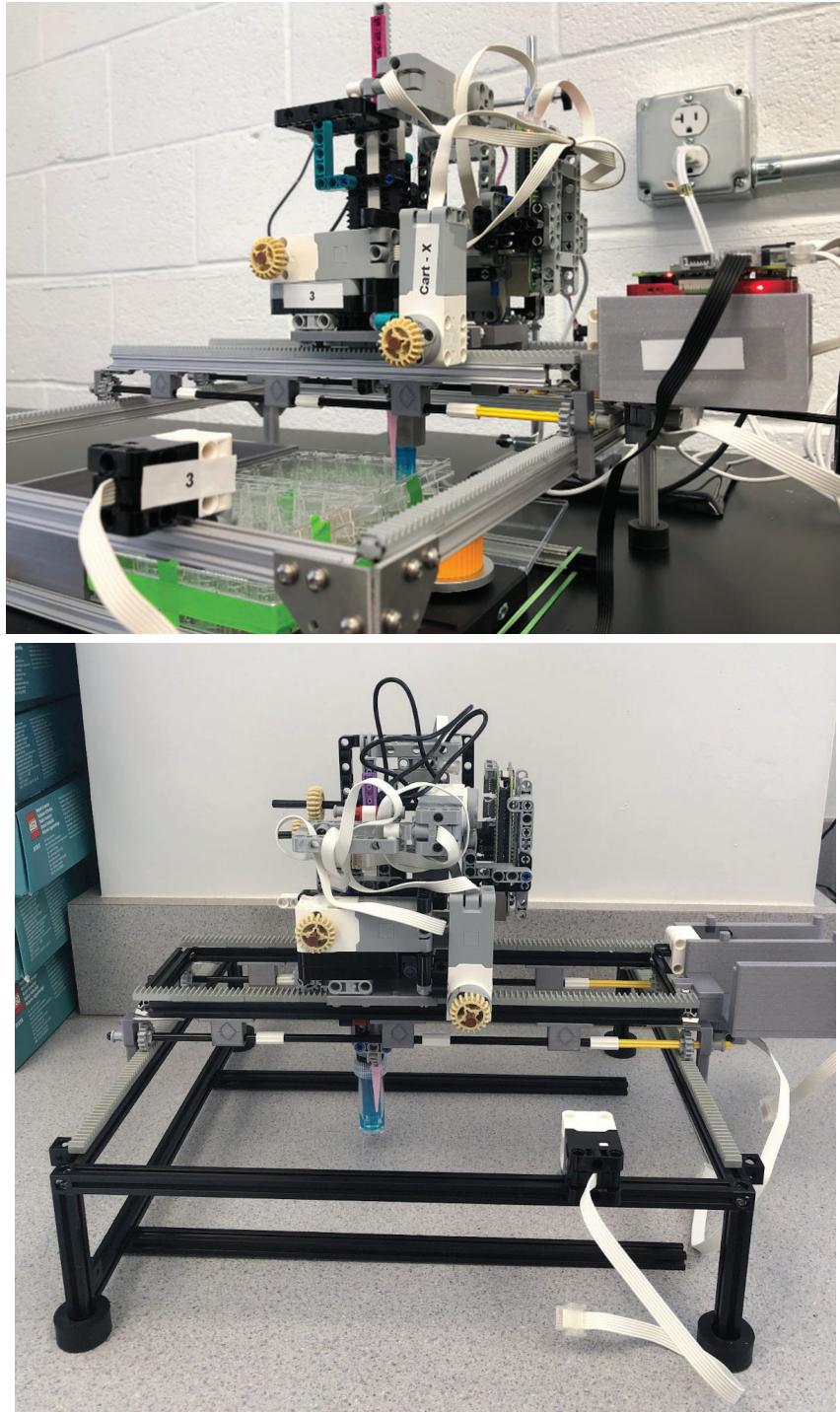


Figure 2.16: (top) LEGOLAS (*Generation IV*) used in the Fall ‘22 UMD course and (bottom) an image of the robot with a black frame (MakerBeam extrusions). This LEGOLAS had more 3D printed components (trolley base, pH sensor lowering assembly, and side cart) that reduced the overall cost of the robot while improving its stability and robustness. Generation IV LEGOLAS could also be equipped with a camera for computer vision based AE (*Chapter 5*).



Figure 2.17: Students working on the LEGOLAS exercises during the Fall ‘22 Implementation, which utilized *Generation IV* LEGOLAS.

Chapter 3: Educational Implementation

3.1 Henderson-Hasselbalch Exercise

The Henderson-Hasselbalch exercise was selected as a simple example for the teaching of AE fundamentals in the *Machine Learning for Materials Science* course at the University of Maryland in the Fall ‘21 and Fall ‘22 semesters as a final project. Students had spent the initial 12 weeks of the class overviewing ML techniques (regression, classification, supervised learning, unsupervised learning, etc.), and had just been introduced to *active learning* about 1 week prior to the start of the project. The selected exercise was *intentionally* simple - being that it contains only one control variable (mixture ratio of acid and base solutions) and 1 response variable (acidity or pH) - so that fundamental concepts could be discussed in conjunction with hands-on implementation. It is fundamentally a chemistry problem, but the ideas of composition-property relationships can be extended into materials science based problems naturally. Although the physical system is simple, layers of complexity can be added with the use of novel active learning techniques (Gaussian Processes, Bayesian Inference, and Entropy Based Acquisition functions) without a related increase in system price. Additionally, the chemical constituents are non-toxic, and safe to work with to eliminate the risk of injury to students.

The Henderson-Hasselbalch equation is a simplified and rearranged mass-action equation that relates the pH of a buffer solution to the relative concentration of an acid and its conjugate base (*Equation 3.1, Figure 3.1*).

$$pH = pK_a + \log_{10}\left(\frac{[A^-]}{[HA]}\right) \quad (\text{Eq. 3.1}) [88]$$

$[A^-]$ = conjugate base conc., $[HA]$ = acid conc., pK_a = log of acid dissociation constant

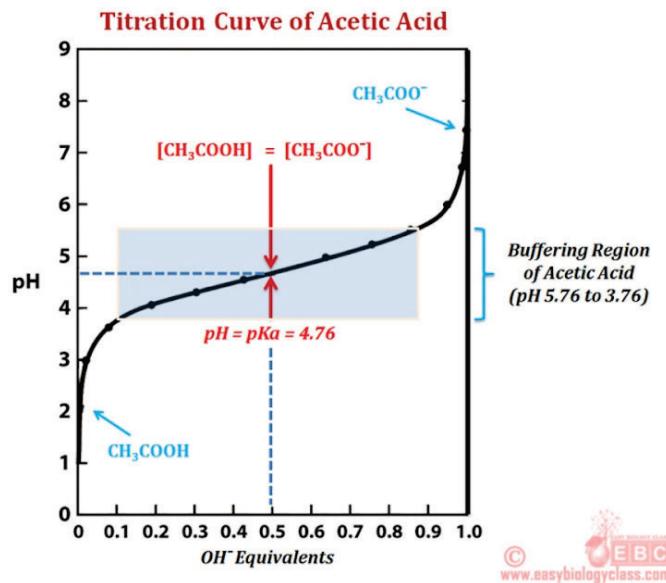


Figure 3.1: The Henderson-Hasselbalch equation for the system of study as expressed in a Titration Curve. The trend here does not appear exactly logarithmic due to the scale of the x-axis (*not in units of [base]/[acid]*) [89]

The equation relies on several assumptions, namely that the acid is monobasic, self-ionization of water may be ignored, the salt base completely dissociates in solution, and the activity coefficient quotient remains constant within experimental conditions [88]. These assumptions are reliably met with buffer solutions of sodium acetate salt (NaOAc) and acetic acid (CH₃COOH) each prepared at 1M concentrations (acetic acid pK_a = 4.756), although the Henderson-Hasselbalch equation begins to deviate from experimental results at high acid compositions due to the violation of certain assumptions in the simplification (*Figure 3.2*) [88]. This deviation from expected behavior is later used as a teaching example of the benefits of non-parameterized models such as Gaussian Processes in materials exploration.

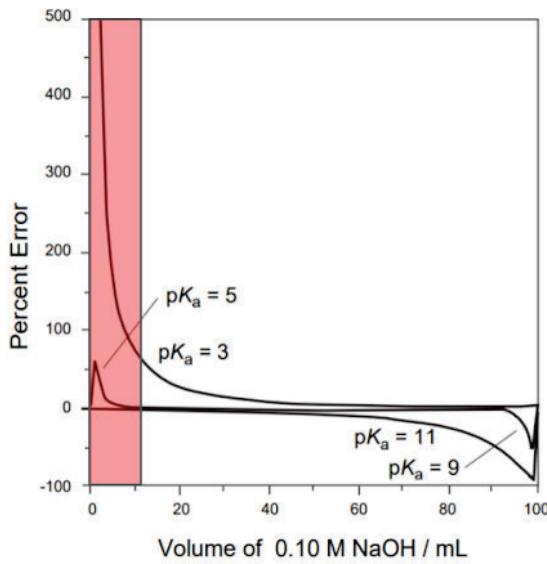


Figure 3.2: Percent error in Henderson-Hasselbalch equation as a function of sodium hydroxide volume per 100 mL of weak acid (see $pK_a = 5$ as a sufficient analogue to our system with $pK_a = 4.756$). The red region indicates solutions of higher acid concentration in which the pH does not correspond well with the Henderson-Hasselbalch equation [88]

The Henderson-Hasselbalch equation represents the functional form of interest in the mechanistic study of the composition-property relationship. In using this example, the key idea to be transferred to students is that the *exploration* of the acidity as a function of composition could aid in accelerating the *exploitation* of a desired pH value (*Section 3.1.2*). In the exercises described in the next two sections, students become familiar with manipulation of LEGOLAS robotics for experimentation, quantify acidity measurement uncertainty, extract a meaningful physical constant of the system (dissociation constant) by fitting a Henderson-Hasselbach equation to the data with non-linear least squares regression, and finally execute a closed-loop Bayesian Optimization based AE by developing a scheduled acquisition function and running the live experiment themselves. Each exercise is intended to build upon skills developed in previous exercises, and to leverage their knowledge of concepts learned in the *Machine Learning for Materials Science* course. The implementation of these exercises - as well as the team-based

nature of the project - are fundamental to the *transferability* aspect of LEGOLAS outlined in *Section 2.1*. This is further discussed in *Section 3.2.3*. Worksheets and Python templates for all exercises may be found at the links below.

Detailed Links: [LEGOLAS System Exercises](#), [LegolasOutline.ipynb](#)

3.1.1 Introductory Exercises

Prior to completing any of the exercises, the students were encouraged to master the system calibration tasks, and peruse LegolasDemo.ipynb, to familiarize themselves with how the robot could be operated. This was a critical step in addressing skills (1), (2), and (3) of transferability: working with their hands on hardware, interfacing with robotics/motors, and demonstrating attention to detail when calibrating the machine. During the implementation of the LEGOLAS as the final project for the machine learning class, it was noticed that groups that tried to forge ahead too quickly to complete exercises without devoting time to these critical steps had much trouble later when trying to run closed-loop AE runs.

Exercise 1: This task asked the students to synthesize one 2 mL sample (50% acid, 50% base), and then measure the pH 10 separate times, cleaning the probe with DI water between each iteration. From these measurements, they were asked to calculate the mean and variance of the pH and present this in the notebook. Completion of this assignment ensured that the group was capable of running the constituent acquisition & deposition functions (by creating the sample), had properly calibrated the liquid-volume/gear ratio (by creating exactly a 2 mL sample), had properly calibrated the pH sensor (by verifying $\overline{pH} = 4.75$), and were giving enough time for the pH measurement to stabilize (~ 20 sec, for a $\sigma^2 < 0.1$). The peculiarities of the measurement

and synthesis system presented here are helpful in addressing aspects (1), (2), and (3) of transferability. It was also important for the students to get some idea of the magnitude of the noise factor in their measurements as they continued to approach the other exercises.

Exercise 2: In this task, the students were given the functional form of the Henderson-Hasselbalch equation, as well as descriptions of the variables therein. They were asked to synthesize a grid of compositions ($\%acid = [10, 20, \dots 80]$, $\%base = 100 - \%acid$), taking pH measurements between each synthesis step (cleaning the probe with DI water again as well). Students were then asked to use a nonlinear least squares protocol to fit *Equation 3.2* to the collected data for the purpose of extracting a pK value (*ideally, pK = 4.75*):

$$Func(x, pK) = pK + \log_{10}(x) \quad (\text{Eq. 3.2})$$

$$x = [\text{acid}]/[\text{base}] = \%acid/\%base$$

While this exercise again required utilizing skills (1), (2), and (3) of transferability, it was specifically suited towards building upon skills (4) and (5); utilizing knowledge of ML techniques, and programming for data manipulation. Using non-linear least squares regression is a critical tool for any scientist dealing with data processing, and - in this example - was shown to produce the “discovery” of a physical constant inherent to the acetic acid buffer system. From the programming perspective, students had to arrange incoming data into an array of values, define a function for optimization, and research the specific nonlinear least squares regression documentation to understand the parameters and returns. It was observed that a small group of students did not understand the reasoning behind alternating between synthesis and measurement (to prepare for an active learning style study) when they could just as easily have prepared all the samples prior to measurement. However, after a discussion, they began to understand the

differences between the format of a grid-like exhaustive study as shown here, and the proposed format of a closed-loop AE study. This was a subtle point of misunderstanding that went apparently undetected during the purely theoretical data-science active learning sessions of the earlier weeks (*i.e., without the hands-on application*).

Exercise 3: In this task, students were introduced to the use of a Gaussian Process and the importance of its kernel and model hyperparameters. There was no experimental use of LEGOLAS within this exercise, as it used the data collected from the grid-like study of *Exercise 2*. Students were asked to use a Radial Basis Function (RBF) kernel, and output the Gaussian Process hyperparameters (kernel length scale, kernel variance, noise variance) after several iterations of a hyperparameter optimizer. Next, they were asked to repeat this process, but now using only 6 data points ($\%acid = [10, 20, \dots, 60]$). As mentioned in *Section 3.1*, the assumptions of the Henderson-Hasselbach equation are valid only within a certain composition range, begetting this limitation of relevant data points (*Figure 3.3*). Students were observed to ask questions about this limitation, leading to further discussions that helped solidify understanding of the flexibility of a Gaussian Process over a rigid parameterized functional form as seen in *Challenge 2*. This discussion briefly addressed aspect (9) of transferability; the discernment to understand pros and cons of AE.

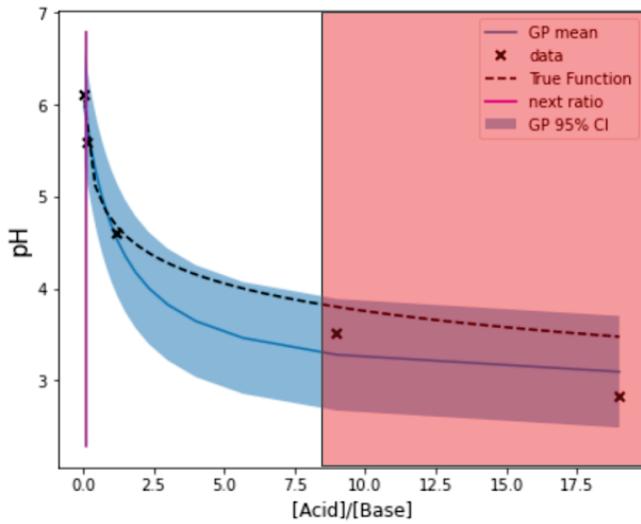


Figure 3.3: Gaussian Process (RBF Kernel) demonstrated for 5 samples within a composition range of %acid = [5,.. 95]. The 95% Confidence Interval of the GP is represented in light blue, with the mean as a dark blue line. It can be seen that the data points (x's) deviate from the expected Henderson-Hasselbach equation (dotted line) at high acid compositions (red region).

In the second portion of Exercise 3, students went further into aspect (9) of transferability, by purposefully implementing bad model assumptions for the Gaussian Process, and analyzing the effects on model outputs. They were asked to rigidly fix certain hyperparameter values - such as length scale - at values that were either too low or too high to see that this could either “underfit” or “overfit” the data. Next, they were asked to use an inappropriate and less flexible kernel (standard periodic) to see that this imposed an (incorrect) periodic form to the Gaussian Process model. Through this type of study, students began to see that these data processing and active learning tools are not necessarily always applicable off-the-shelf, and could sometimes require *human insight* or tuning to be of real service. Throughout this exercise, students were also asked to plot the Gaussian Process (mean, 2σ Confidence Interval), idealized Henderson-Hasselbalch model (for reference), and experimental

data (*Figure 3.4 for example graphs*). This ensured that they were also working on skill (8) of transferability; presenting data in a digestible & meaningful fashion.

3.1.2 Autonomous Closed Loop Exercise

The Autonomous Closed Loop exercise was the pinnacle achievement for the students and was intended to usurp the most time & thought on their part. It was designed to combine all 9 critical aspects of transferability, with emphasis on (6), (7), and (9): the insight to encapsulate research “objectives” into active learning schema, the foresight and knowledge to simulate live-runs, and the discernment to understand AE pros & cons. Students who did not have a good grasp of the theoretical fundamentals of active learning (from the prior classwork) struggled to understand the significance and development of acquisition functions. Many students also had a hard time differentiating between what *they* knew about the composition-property relationships (because they were given the Henderson-Hasselbalch equation) and what *the robot* knew (because in this toy problem we were assuming it began uninformed) [66]. This was surprising, as these same students had succeeded in completing simulated active learning assignments earlier in the course. However, it highlighted the most beneficial aspect of LEGOLAS: *AE as a field cannot be taught reliably and effectively without a real-world tangible system for experimental application*. The majority of these same students that struggled to understand the benefits of active learning and AE were then capable - after successfully completing *Exercise 4* - of clearly elucidating their thought processes and acquisition function implementation while presenting their final project results.

Exercise 4: In this task, the students were able to use the Gaussian Process understanding and code they developed in *Exercise 3* within a closed-loop AE process framework to *autonomously*

determine the composition (mixture ratio) at which the pH = 4.75. As stated earlier, this led to some confusion among students, as *they* already knew what composition this occurred at (50% acid, 50% base), *and* they had already collected enough data in Exercises 1-3 to reliably extract this. It was also confusing to students that this was indeed an *exploitation* based CO, because the property value they were targeting was neither a maximum nor minimum acidity metric. The students were told to discuss the merits of different acquisition function schemas, and encouraged to combine *exploitation* and *exploration* COs in some manner. This was a reiterative process in which students would come to the TA's for guidance. Collectively, the class would first talk through what it *meant* to explore (looking towards the *unknown*), what it *meant* to exploit (where do we think our target is considering what we know?), and what would be an appropriate order in which to execute those initiatives. The drawbacks of exploiting from the start (i.e., “shooting in the dark”) and purely exploring (effectively a grid search) were discussed. A question that often came up was *why* we were doing this in the first place. (The goal is to reduce the number of experiments). Of course, a toy problem like the Henderson-Hasselbalch equation seemed to not need this reduction in experiments, but the discussion of the merit of AE in this case was a valuable expression of aspect (9) of transferability; discerning between AE pros and cons.

Once the theoretical and philosophical discussions of the CO order were understood, the students again came for guidance with respect to the mathematical encapsulation of these *human ideas* into quantitative acquisition functions. This discussion, along with the prior philosophical one, embodied aspect (6) of transferability most directly; the insight to encapsulate research “objectives” into active learning schema. Some groups decided on a hardcoded cutoff between a purely exploration based CO (*Equation 3.3*) and purely exploitation based CO (*Equation 3.4*).

This could, for example, consist of doing 4 experiments exploring, then spending the remainder of the experiments exploiting. Other groups understood more fundamentally that although this approach could work, it fundamentally depended on our *prior knowledge* of the simplicity of the system for picking this cutoff point. Those groups, instead, utilized their knowledge of other techniques - namely, a modified upper confidence bound (UCB) - to have an adaptable and continuous transition between these COs (*Equation 3.5*)[90]. Certain aspects of UCB tuning led to further discussions, broadening the students' understanding of the potential data research opportunities within the acquisition function based side of AE.

$$x = \operatorname{argmax} \left[\sigma_{GP}^2 \right] \quad (\text{Eq. 3.3})$$

$$x = \operatorname{argmax} \left[- |\mu_{GP} - 4.75| \right] \quad (\text{Eq. 3.4})$$

$$x_n = \operatorname{argmax} \left[(- |\mu_{GP} - 4.75|) + \sigma_{GP} \left(c_1 \sqrt{\frac{\ln n}{n}} \right) \right] \quad (\text{Eq. 3.5}) [90]$$

n = sample number, c₁ = tunable coefficient

In the actual implementation of *Challenge 4*, the students also began to understand that they needed an *exit criterion* at which the AE loop would terminate. Upon further discussions, it became clear that this could introduce even more complexity to their AE loop. If, by chance, they serendipitously selected a composition of 50% acid, 50% base (with a pH sufficiently close to 4.75) during their *exploration* initiative portion, would that constitute success? Should they verify their result? Was it important to continue searching the composition space for other potential compositions that might produce this specific acidity? No singular answer to these questions was identified as correct, and the discussion itself was the teaching point. *Without a tangible experimental system such as LEGOLAS, pertinent questions like this are easily overlooked.*

As mentioned before, some groups who did not give much attention to calibration (*aspect (3)*) either had hardware malfunctions with the robot (they are not supposed to touch or assist the robot during the final closed-loop study), or erroneous pH metrics that led to incorrect answers. Another critical aspect of designing their final closed-loop AE was to leverage the utility of simulated runs. Being that the synthesis/measurement/analysis loop could take an average of 90 seconds, groups that attempted to troubleshoot their active learning code during live runs could become extremely frustrated and get far behind others due to the time they were spending resetting the experiment. Those that followed the advice of using simulated experimental data (i.e. a Henderson-Hasselbalch function with some random white noise applied) refined their active learning code, and once they knew it worked, were able to confidently proceed with experiment (*aspect (7) of transferability*).

An example of a live-run evolution of the Gaussian Process for a purely exploration based CO (*Equation 3.3*) is shown in *Figure 3.4*, for samples 1 through 5. The Gaussian Process can be seen to approach the logarithmic behavior of the Henderson-Hasselbalch equation, as the acquisition function prioritizes compositions with high degrees of model uncertainty.

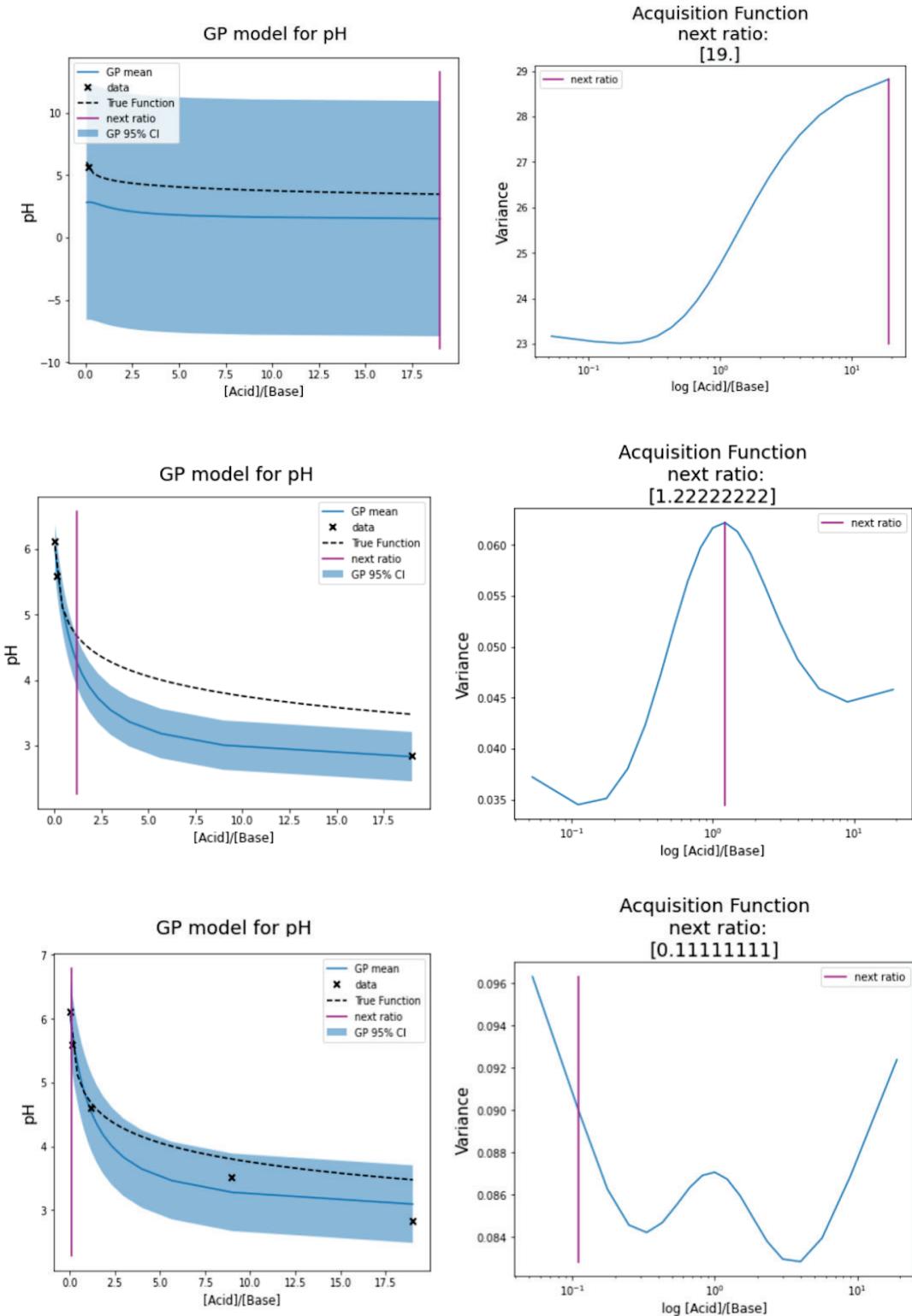


Figure 3.4: Gaussian Process evolution (RBF Kernel) for first 5 experimental samples with a purely exploration based acquisition function.

3.2 Class Implementations

This section provides a summary of the implementations of LEGOLAS within the *Machine Learning for Materials Science* course in the Fall ‘21 and Fall ‘22 semesters. This class (3 credits) is listed on the University of Maryland Schedule of classes as ENMA437 (*previously ENMA 489L*) for undergraduates, and ENMA637 for graduate students. It has been offered once a year in the Fall semester since 2020. The enrollment has on average been around 12-16 students, typically half graduate and half undergraduate students. I have served as the main teaching fellow for the class in 2021 and 2022, when LEGOLAS was fully implemented as the final project tool. Prior to the Fall 2021 offering of the class, I had spent the summer developing LEGOLAS.

Course Description: *Familiarizes students with basic as well as state of the art knowledge of machine learning and its applications to materials science and engineering. Covers the range of machine learning topics with applications including feature identification and extraction, determining predictive descriptors, uncertainty analysis, and identifying the most informative experiment to perform next. One focus of the class is to build the skills necessary for developing an autonomous materials research system, where machine learning controls experiment design, execution, and analysis in a closed-loop.*

The LEGOLAS Exercises were designated as the final project for the class, and students were split up into teams of 4-6 per robot. They were given the last 3-4 weeks of the in-period class time (3 hrs/week) to work on the project, and allowed extra lab time if they felt they needed it. Credit was given for each exercise either through TA observations of completion, or through a video taken by group members (*see LEGOLAS rubric in Detailed Links*). Additionally, students prepared a final presentation (20 minutes) in which they overviewed their completion of

the Exercises, elucidated their decision making processes, highlighted the difficulties they had, and offered suggestions for course improvements.

Detailed Links: [LEGOLAS Rubric](#)

3.2.1 Fall 2021

In the Fall of 2021, the class utilized *Generation I & II* LEGOLAS systems (*Figures 2.12 - 2.14*). There were 3 teams of 6 students, with 1 graduate and 2 undergraduate groups. The graduate group was assigned to the *Generation II* LEGOLAS because the other two EV3 based Generation I devices were easier for undergraduates to connect to and operate (used a bluetooth connection). During this implementation, it was clear that mechanical improvements needed to be made to the system, with parts often failing and requiring TA attention. It was also clear that connectivity instabilities were present with both Generation I & II LEGOLAS, and a new interface would need to be established for further classes. Additionally, groups of 6 proved too large (*Section 3.2.3*). Only one group (graduate) was able to experimentally finish *Exercise 4*, while the other groups were given partial credit for running a simulated version of it.

3.2.2 Fall 2022

In the Fall of 2022, the class utilized *Generation IV* LEGOLAS systems (*Figures 2.16, 2.17*). There were now 4 groups of 4 students, with 2 graduate and 2 undergraduate groups. During this implementation, the teams suffered less from hardware issues, but did have some troubles with using the calibration GUI and expressed frustration with the excess time it took to learn how to use it. Two groups (one graduate and one undergraduate) were able to successfully finish *Exercise 4*, while again the other groups received partial credit for simulated runs. It was observed that many teams spent excess time (*up to 3 weeks in some cases*) on the initial exercises, and did not properly gauge the increased workload of *Exercise 4*. Many of these

groups had to spend 10-15 hours of additional time outside the designated class-time to finish their exercises. Therefore, in the future, the teaching faculty aims to emphasize a proper timeline to encourage completion of initial exercises by earlier checkpoint dates.

3.2.3 Lessons Learned

From the two class implementations of LEGOLAS, several lessons were learned that allowed for improvements in the robot design and pedagogical style. These involved the complexity of coding templates, software and hardware issues, optimal group size and resulting dynamics, use of checkpoints/deliverable dates, and robot POV issues.

Coding Templates: In Fall of ‘21, the students were provided a thorough template in Jupyter Notebooks that included all the experimental control functions and a “framework” for the closed-loop structure of *Exercise 4*. All students had to do was input the mathematical form of the acquisition function and then the exercise could be completed (provided they had calibrated the robot and pH sensor properly). This ease of execution ultimately led to a lack of practice with data structure management, and additionally appeared to hinder fundamental understanding of the active learning concepts (as observed in the final presentations). Therefore, in the Fall ‘22 implementation, although the students were still provided a resource for sourcing robotic control functions (LegolasDemo.ipynb), they were given little to no framework (LegolasOutline.ipynb) and had to develop both data structures and the closed-loop workflow from scratch. The need to self-develop the workflow proved instrumental in driving home the iterative synthesis/measurement/analysis concepts, but it was also noted that many groups struggled with data structure management, likely due to a lack of proficiency in Python. UMD recently implemented a mandatory Python course for undergraduates, and this is expected to ameliorate

some of these issues. Nonetheless, it may be helpful in future implementation to contain more “hints” and suggestions that could point students towards the proper solution framework, without explicitly determining it for them beforehand.

Hardware/Software Issues: *Generation I & II* LEGOLAS contained no 3D printed components, and therefore showcased mechanical instabilities at LEGO/Aluminum joints that proved difficult for students to handle. Additionally, Bluetooth connectivity to the EV3 Brick was unstable at times, and led to frustration among the students and TAs that detracted from the important aspects of the course. *Generation IV* LEGOLAS was developed to improve stability and reduce these hardware issues. By providing manual controls for all the motors (tan gears), a more stable interface mechanism for LEGO/Aluminum joints (3D components), and a simplistic GUI for calibration, students experienced less hardware issues. However, students expressed frustration with the time involved in calibrating the robot, and there were still some networking problems (WiFi connectivity issues). Although it is important for the students to take care in managing the hardware and calibration of the robot (*aspects (1-3) of transferability*), the research group is taking steps to ameliorate the majority of these problems for future implementations.

Group Size and Dynamics: The Fall ‘21 implementation featured 6 students per LEGOLAS robot. As is the case in large groups, large inequalities in member participation were observed, and usually 1 to 2 members of each group did the majority of the work or came in after class to finish exercises. Therefore, in Fall ‘22, more LEGOLAS were produced to facilitate groups of 4 students. Although this promoted group better member participation equality, it was observed that students sub-divided themselves to tackle different aspects of the exercises. Some students

would handle the coding aspect, while others worked on calibration and experimental setup, and yet others on acquisition function development. This appears to be a natural result of group dynamics, and is not in itself a negative aspect. However, to fully satisfy the MGI emphasized foundational pillars (data/computation/experiment) it should be a goal of further implementations to test individual members across all foundational categories to ensure they are understanding them at least at the conceptual level. This is currently hard to assess in the format of a group submission and group presentation, where students can focus solely on their category of expertise to achieve a good grade.

Project Timeline and Checkpoints: In both implementations, students were observed to *underestimate* the amount of time calibration, experiment, and troubleshooting would take. Most groups did not show initiative in prudently finishing the introductory exercises (*Exercises 1-3*), because they believed that *Exercise 4* would take the same amount of time. In further implementations of LEGOLAS, the teaching faculty will enforce checkpoints and deadlines to a greater extent to help students avoid this mistake.

Robot POV: It was clear that *Exercise 4* provided conceptual issues to some students, usually due to their inability to distinguish between what the robot was intended to know (i.e. nothing at the start), and what they knew (i.e. the Henderson-Hasselbalch equation, and therefore, which composition to target for a pH of 4.75). This appears to be inherent to autonomous teaching examples, but one way the teaching faculty could ameliorate this is to elaborate upon Robot POV more in the problem definition of *Exercise 4* [62].

Chapter 4: Autonomous Model Exploration

This section contains information regarding the use of Bayesian Optimization, Bayesian Inference, and an informational entropy based acquisition function for Autonomous Model Exploration. In each section, detailed links are provided to Jupyter Notebooks containing the code used to produce the live closed-loop results using LEGOLAS. All of these ML and active learning modules are intended to be instructional in manner and could be implemented into the classroom setting using LEGOLAS, if desired.

4.1 Bayesian Optimization

The Bayesian Optimization (BO) module for LEGOLAS focuses on the use of a Gaussian Process (GP) to construct an acquisition function that prioritizes exploration, exploitation, or a (scheduled) mixture of both COs (*Section 1.3.1*). The relevant Bayesian Statistics background needed to understand GPs and BO is included in *Section 1.3.2*. In this section, GPs and BO will be briefly explained in terms of their application to the Henderson-Hasselbalch problem.

Prior to Data Collection: A GP is a non-parametric probabilistic supervised ML method that can be used for either regression or classification [37]. In the case of the Henderson-Hasselbalch exercise, it is utilized for regression: fitting a model to the data. We consider a discrete set of n compositional values ($\%Acid$) for which we will be predicting the expected response (pH), and construct an n -dimensional multivariate normal (*gaussian*) distribution [37]. For construction of the GP, we will treat each potential composition as the i -th random variable in this n -dimensional multivariate normal, and will sample across this prior distribution to create many potential functions (*Figure 4.1*). This prior distribution is initially defined by a mean (usually assume

$\mu = 0$ to simplify equations), and a *covariance matrix* ($n \times n$ dimensional), which sets the shape of the distribution by declaring the absolute variance for each random variable as well as the metrics of covariance *between* each random variable [37]. The covariance matrix is then in turn defined by a kernel, which is a functional form that determines the covariance between random random variables, and exerts the influence of this relationship relative over a user-defined length-scale (in units of distance between random variables). Therefore, the functional form of your potential functions and their “smoothness” is determined by your selection of a kernel [37]. *Figure 4.1* shows how the sampling of potential functions across the prior distribution produces a mean of 0 across the composition space, and a uniform variance (shown by the 95% confidence interval). *Figure 4.2* shows the probabilistic interpretation of this (regarding the likelihood of the model) that will be used to construct an uncertainty landscape once we begin collecting data.

In *Exercises 3 & 4* for LEGOLAS, the students use a hyperparameter optimizer to tune the length scale and other GP attributes, and then are asked to select inappropriate kernel types (periodic, linear, etc.) and parameters to better understand the deleterious effects this can lead to in regression (overfitting, underfitting, etc.). In the Henderson-Hasselbalch study, a stationary and flexible Matérn-5/2 Radial-Basis Function (RBF) kernel was utilized to explore the composition/response landscape [24]. For a more in-depth discussion of Gaussian Processes, please see this [source](#) [37].

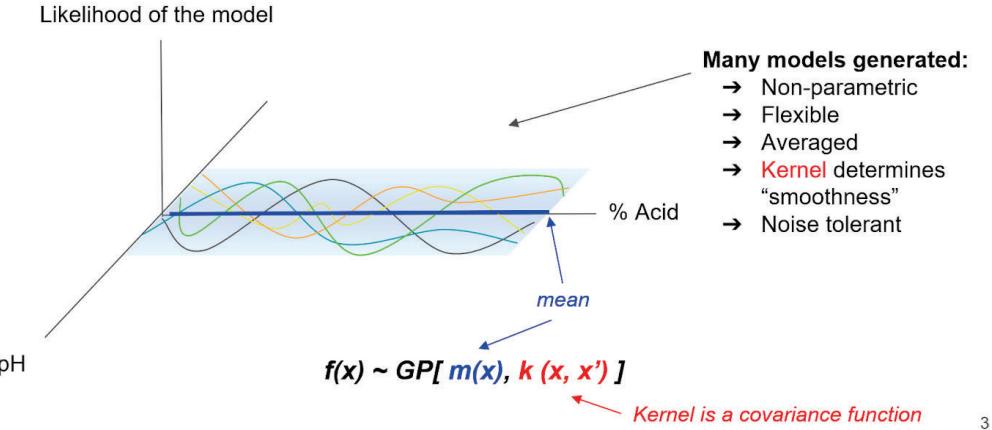


Figure 4.1: Sampling of potential models across the prior distribution determined by the kernel

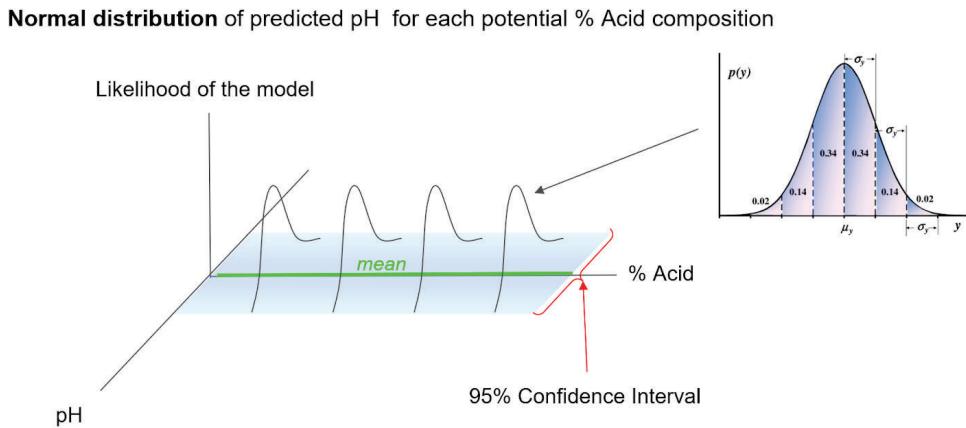


Figure 4.2: Probabilistic interpretation of the GP for uncertainty quantification and propagation

During Data Collection: Once data is collected, the potential functions composing the GP are then constrained to pass through this point (*Figure 4.3*). The sampled functions may also be only required to pass within a certain range of the measured data (of a magnitude set by the user-defined measurement error) [37]. This concept shows the importance of incorporating *Exercise 1 (measuring measurement uncertainty)* in properly understanding and applying GPs. Upon constraining the GP near the acquired data, the variance distribution across composition space is no longer uniform, and can be utilized in an exploration based acquisition function to

guide an AE loop (*Figure 3.4*, *Equation 3.3*). Alternatively, the mean of the GP may be used to perform an exploitation based AE loop (*Equation 3.4*). These two techniques are what constitute the use of BO in exploring the Henderson-Hasselbalch example system.

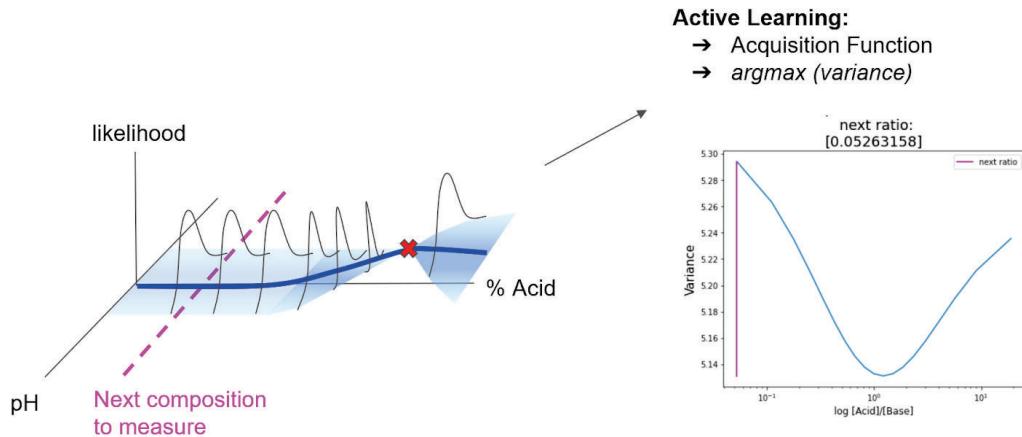


Figure 4.3: (left) Effect of new data on GP structure, and (right) purely exploration based acquisition function

4.2 Hypothesis Validation Objective

One of the main questions this thesis aimed to consider was how an autonomous system could perform within the role of a lead scientist: a role which could be ambiguous, dependent on insight, and itself a method of discovery. At a high level, the scientific process can be broken down into observation, hypothesis, and experiment. It has been shown that in the realm of experiment, automated systems can offer an improvement of reliability, reproducibility, and even efficiency [7,29,31]. However, the heart of this debate lies in the interrelated processes of observation and hypothesis, which - at least in the case of breakthrough discoveries and groundbreaking science - appear to rest upon an ineffable characteristic of human *insight* [5,6,9].

Is it possible for autonomous systems to approximate this insight, or are they forever condemned to execute only the experimental duties?

Observation and hypothesis both rely on some assessment or attribution of *value* to certain observables within the expanse of potential observables for any system of study. It is the selection of *valuable* or relevant observables that leads to the generation of a hypothesis - also selected amongst a vast number of potential hypotheses - and the validity of this hypothesis then relies upon the design of the necessary experiment and its ability to tell us only if it is *wrong* [91]. In the scientific process, lead scientists are never concluding that their hypothesis is *correct*, but rather are forever assessing the ability of their hypothesis to agree with as many observables as possible [91]. At the same time, in accordance with Occam's Razor, they are searching for as simplistic a hypothesis as possible [91,92].

In our point of view, then, if we first restrict ourselves to a set of human defined observables of importance (i.e., our response metrics), then an autonomous system approaching the task of a lead scientist would have 3 major tasks from there:

- (1) Generate many potential hypotheses from these observables
- (2) Assess (or rank in terms of confidence) these hypotheses with respect to their ability to explain the observables and their simplicity
- (3) Design the next experiment to best help differentiate between our potential hypotheses -i.e. help us increase our confidence in one select hypothesis

If these steps are executed iteratively in conjunction with automated control of experimental prerequisites, the result would then be an autonomous scientist concerned with establishing an explanatory hypothesis of the greatest simplicity and fidelity in a minimum number of experiments.

In this section, an active learning schema is developed to achieve these 3 main goals. It is applied to the simple Henderson-Hasselbalch experiment described in *Chapter 3* for autonomous live-runs. In one case, a list of user selected hypotheses are initially proposed, for which the robot's task is to determine which one best explains the data and with the greatest simplicity (*Step 1*). In the 2nd case, symbolic regression - also known as genetic programming - is used to iteratively generate and update the list of these potential hypotheses, and the robot's task is to both generate the correct hypothesis (the Henderson-Hasselbalch equation) and select it among the other potential functional forms (*Step 1*). In both cases, the potential hypotheses are iteratively evaluated along the metrics of fitness and simplicity to create a ranking akin to the robots confidence in them (*Step 2*). Cumulative distributions composed of all hypothesis predictions across the search space are generated (*where each model's contribution is weighted by its ranking from Step 2*). The composition corresponding to the cumulative distribution with the highest informational entropy (i.e. the greatest disorder among predictions) is identified as the best next-experiment for the purposes of distinguishing between the relevant candidate hypotheses and increasing the robot's confidence in one as the explanatory function.

Section 4.2.1 describes the statistical basis of this loop in more detail, and *Section 4.2.2* describes the results when applied in an autonomous live-run scenario. It is shown that in both cases, the robot is able to generate and identify/validate the correct hypothesis within 5-6 experiments. The ability to inexpensively perform live-run experimental testing of acquisition functions is a testament to the usefulness of having a low-cost modular SDL available to a research group or educational environment. Students and researchers can then easily develop, test, and improve their active learning ideas in a real-life experimental setting.

4.2.1 Generating and Evaluating Candidate Hypotheses

Preset Functional Forms

To test the hypothesis validation aspect of the active learning schema, it was first utilized with a list of user defined preset functional forms. Each of these functional forms represents a potential hypothesis, which will be called a “candidate” from here on out. The seven selected forms relied on parameters (A,B,C,D) that could be optimized to fit the data using non-linear least squares regression after each pH measurement was performed in the closed-loop (*Equations 4.1*):

$$\text{Logarithmic} - f_1 = A - B * \log(x) \quad (\text{Eqs. 4.1})$$

$$\text{Exponential} - f_2 = A + B * \exp[C * (x - D)]$$

$$\text{Linear} - f_3 = A + B * (x - C)$$

$$\text{Quadratic} - f_4 = A + B * (x - C)^2$$

$$\text{Cubic} - f_5 = A + B * (x - C)^3$$

$$\text{Sinusoidal} - f_6 = A + B * \sin(C * x)$$

$$\text{Power Function} - f_7 = A + B * (x - C)^D$$

$$\text{where } x = [\text{Acid}] / [\text{Base}]$$

After each candidate was fit to the data, they could be ranked - under certain assumptions - along their Bayesian Information Criterion score (*Equation 4.2*), which took into account their complexity (based on the number of parameters needed for each candidate) and the candidate’s error variance (a metric of their fitness) [93]:

$$BIC = N * \ln\left(\widehat{\sigma_e^2}\right) + k * \ln(N) \quad (\text{Eq. 4.2}) [93]$$

where N = # of measurements, k = # of model parameters

$$\left(\widehat{\sigma_e^2}\right) = \text{error variance}$$

The BIC penalizes candidates with a greater number of model parameters (k) and rewards those with a lower error variance (i.e. a better fit to the data) [93]. The model with the lowest BIC was then considered to be the leading candidate according to the robot (*Step 2*). The ultimate goal of this closed-loop experiment was to have the robot identify the logarithmic equation (with $A = 4.75$ & $B = 1.0$ corresponding to the Henderson-Hasselbalch equation) as the candidate with the lowest BIC score after just a few experiments. The experiment, candidate fitting, and candidate ranking steps can be visualized in *Figure 4.4*.

Symbolic Regression for Candidate Generation:

A more advanced task for LEGOLAS was then to autonomously generate potential hypothesis candidates prior to ranking them. In this case, an open source Python specific symbolic regression software (PySR) was utilized to propose candidates that fit the data. These were then assessed again by complexity (a function of number of parameters and functions) and their mean squared error (MSE - also a metric of fitness) to produce the ranking system (*Figure 4.4*). Only the top 5 candidates would be considered for use in the acquisition function (*Section 4.2.2*). It was observed that the symbolic regression software generated some simple underfitting models (linear, constant), and other more complex overfitting models (utilizing both logarithm and sinusoidal functions), but the best ranking eventually was attributed to the purely logarithmic function (within 5-6 experiments).

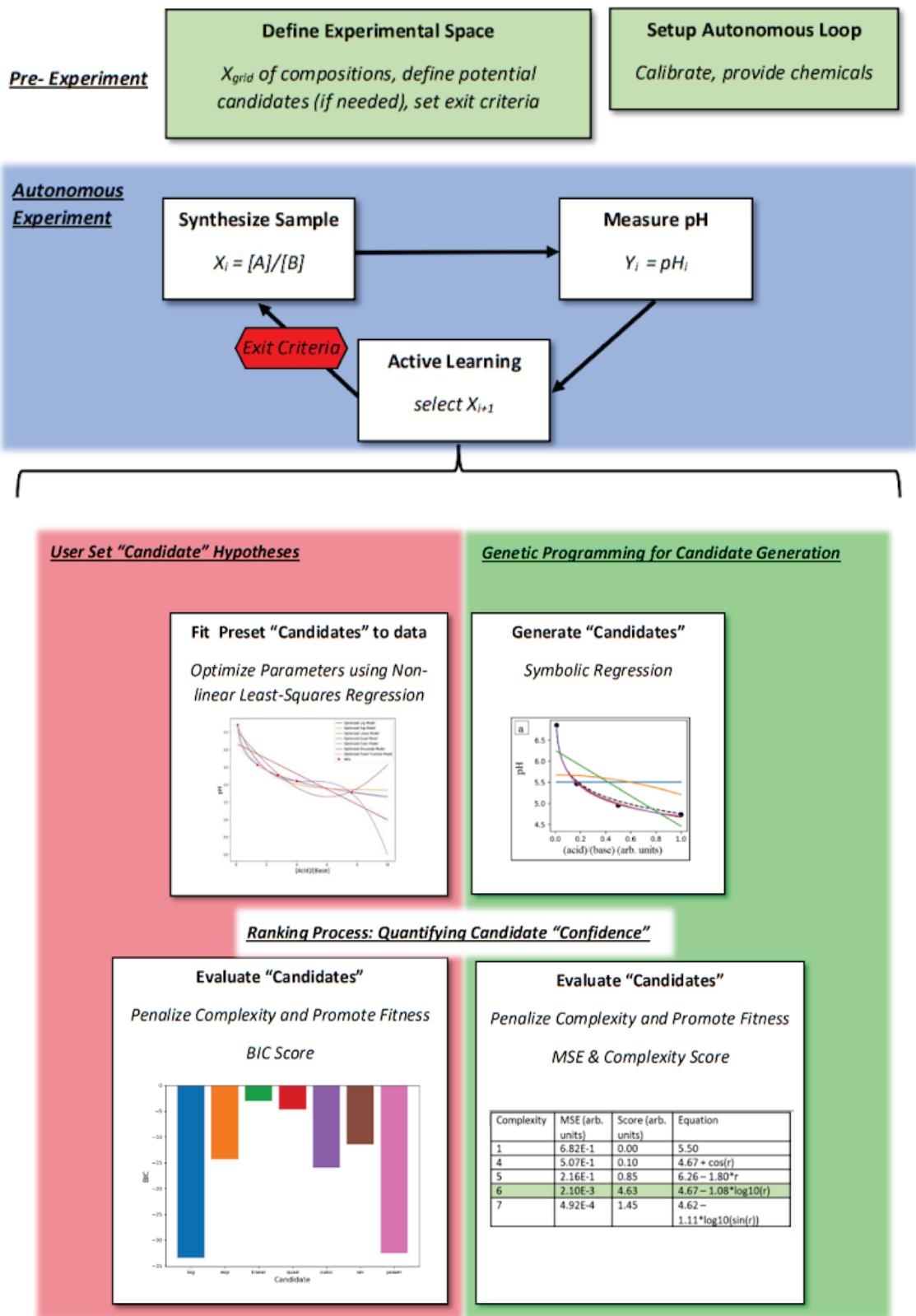


Figure 4.4: Process-flow for experimentation, hypothesis generation, and evaluation

4.2.2 Informational Entropy Acquisition Function

To accomplish *Step 3* (designing the next experiment to best help differentiate between our potential hypotheses) an acquisition function involving informational entropy was utilized. First, a uniform gaussian error distribution was assumed along the entire sample space for each candidate, essentially comprised of a normal probability density function (PDF) with characteristic width determined by the calculated standard deviation of the candidate (*Equation 4.3*):

$$\text{Candidate PDF} \left(y; \mu_i, \sigma^2 \right) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left[\frac{y-\mu}{\sigma}\right]^2\right) \quad (\text{Eq. 4.3}) [94]$$

$$\text{where } \sigma = \sqrt{\frac{\sum(pH_i - \mu)^2}{N-1}}$$

μ is the mean (functional form prediction)

y = acidity (pH)

pH_i = measured/observed mixture acidity values

and N = number of measurements

The standard deviation of each candidate, a measure of how well it fits the observed data, would then determine the width of these normalized PDFs. A bad fit would have a wide distribution, while a better candidate would have a narrow one. These candidate specific PDFs were then summed into a cumulative distribution - a sort of Gaussian Mixture Model (GMM) - and normalized so that the area under the distribution was equal to 1 (so that this could be used as a cumulative probability density function) (*Equation 4.4*). This was repeated at each mixture value in the grid of mixtures that were considered (300 [acid]/[base] compositions):

$$GMM_i = \sum_{q=1}^m \left(\left[\frac{1}{m} \right] PDF_q \right) \quad (\text{Eq. 4.4})$$

where q = candidate index, i = mixture index, m = # of candidates considered

GMM_i is the Gaussian Mixture Model for mixture i

PDF_q is the Probability Density Function for candidate q

Due to the inherent normalization of the PDFs prior to creation of the GMMs, it was the case that “poor” candidates - with their wider width and lower profile - did not affect the overall qualitative shape of the GMM as much as “good” candidates did. Later on, when selecting the GMM with the highest informational entropy, this characteristic led naturally to a greater weighting of importance to discrepancies between predictions of only the best candidate models.

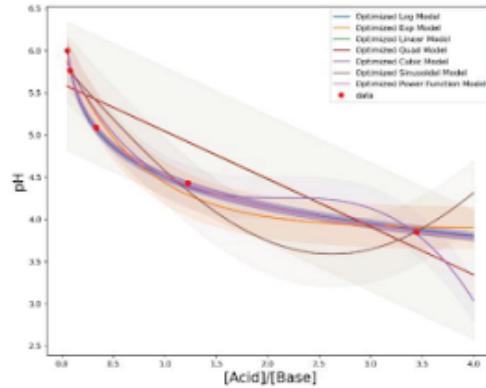
The cumulative PDF creation process can be visualized in *Figure 4.5*.

Create Cumulative probability density functions for the response variable

Produce PDF for each Candidate

Assume gaussian error distribution or noise

Fitness affects distribution (STD based on MSE)



Create Gaussian Mixture Model (GMM_i)

Add up Candidate PDFs to create a CDF at each X_i in X_{grid}

Worse candidate PDFs have smaller "impact" on CDF shape

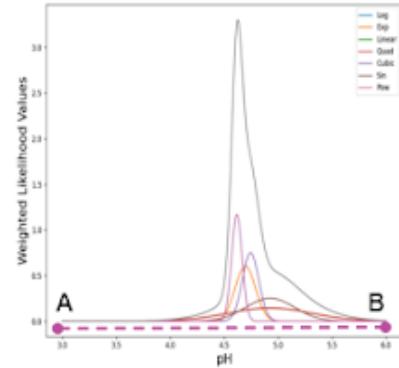
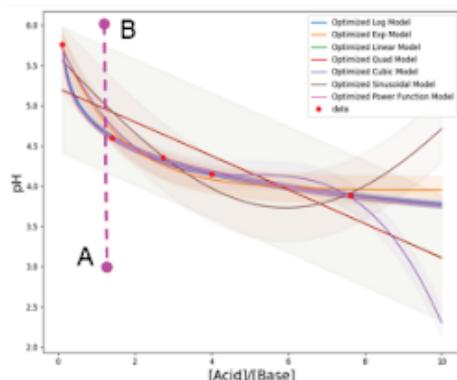


Figure 4.5: Process-flow for creating GMMs over the sample space

The last step of the acquisition function was then to select the GMM with the greatest informational entropy. The informational entropy, or Shannon Entropy, can be formally calculated from *Equation 4.5* [26]. Because we did not use an analytical form for the CDF, the probabilities necessary to calculate Shannon Entropy were then instead numerically estimated using *Equation 4.6* via integration along a grid of 300 discrete pH values in the response space.

$$S = -K \sum_{i=1}^n p_i \log(p_i) \quad (\text{Eq. 4.5}) [26]$$

where K is a constant related to the units desired

$$S_{GMM_i} = -\sum_{j=1}^{300} p_j \log(p_j) \quad (\text{Eq. 4.6})$$

where $p_j = \int_{y_j}^{y_{j+1}} (GMM_i) dy$

Informational entropy has been used in several active learning search algorithms for its use as a metric of uncertainty or disagreement [95-99]. The use of informational entropy in our problem to quantify the level of disagreement or uncertainty in candidate predictions follows naturally from Shannon's paper in which he notes that the entropy will be at a maximum if all p_j are equal [26]. Therefore, the composition mixture at which the entropy is the highest should correspond to the area of highest uncertainty between our best candidate predictions, and would naturally be a good next-experiment if our goal is to differentiate between our candidates and select the one we are most confident in. The process of calculating this entropy and using it to select the next composition to measure is shown in *Figure 4.6*.

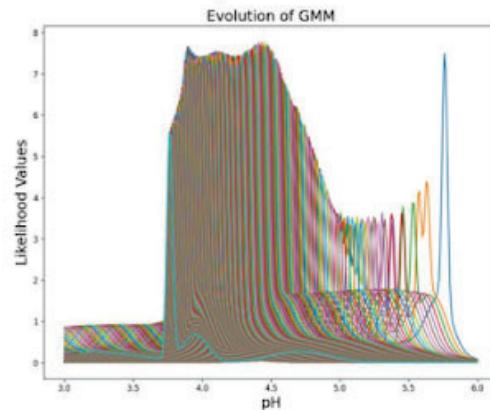
Using Informational Entropy as an Acquisition Function

Estimate Informational Entropy for each GMM_i

Use numerical integration/summation over discrete X_{grid} & Y_{grid}

$$S_{GMM_i} = - \sum_{j=1}^{n=300} (p_j \log p_j)$$

$$p_j = \int_{Y_j}^{Y_{j+1}} (GMM_i) dY$$



Select Next Composition based on Maximum Entropy

GMM with highest S_{GMM_i} shows experimental conditions for greatest disagreement between best models

$$X_{i+1} = X_{\text{argmax}[S_{GMM_i}]}$$

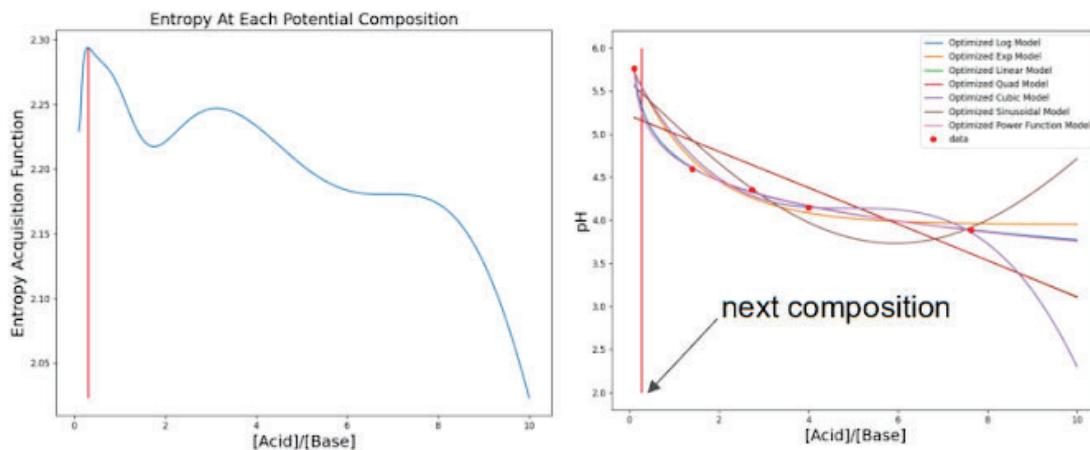


Figure 4.6: Using the informational entropy metric to select the next composition

Chapter 5: Future Work and Conclusions

5.1 Future Work & Scope

5.2.1 Alternate Educational Exercises

LEGOLAS may easily be expanded beyond the exercises described in this thesis due to its modular 3D printed design, LEGO components, and room for additional experimental hardwares. This section describes some alternative uses for LEGOLAS that are safe for classroom adoption and simple enough for educational purposes. They rely on camera vision for characterization and calibration. This functionality can easily be added to LEGOLAS via a USB compatible camera on the bottom of the Trolley cart (*Figure 2.2*).

Prior to the authors introduction to the project, LEGOLAS had been used for color mixing studies. The camera was used to assess the color of prepared dye solutions (combinations of blue, yellow, or red) to learn the color mixing rules and then, if desired, produce a certain colored solution (*exploitation CO*) on command. In addition to building on computer vision skills, this exercise also incorporated clustering techniques for identifying the location of sample wells and adjusting the trolley's position in order to remove the need for hard-coded calibration procedures (*Figure 5.1*).

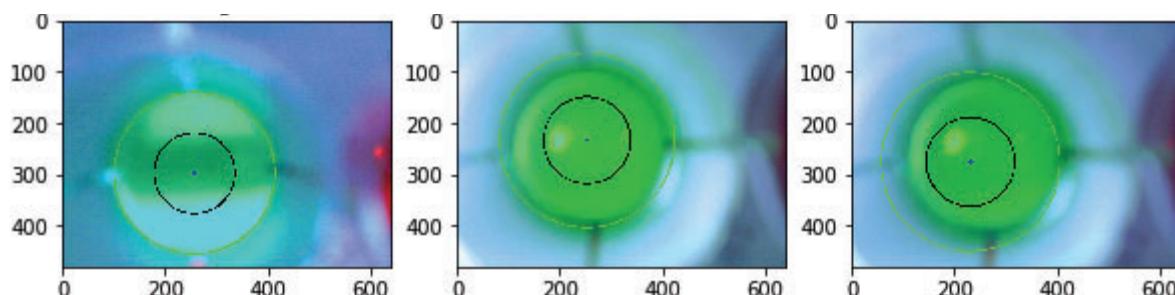


Figure 5.1: Camera images of sample wells from the color mixing study

Another potential exercise that the current LEGOLAS could be used for is that of the Iodine clock reaction. This type of reaction only requires widely available chemicals and household products (tap water, vitamin C tablets, iodine tincture, hydrogen peroxide, and liquid laundry starch) to complete [100]. The time delay between the addition of all chemicals and the color change of the solution is a function of initial reactant concentrations as well as temperature, providing an opportunity for an active learning based approach to learning this relationship and exploiting a particular time delay [101].

5.2.2 System Modifications

To expand beyond the potential exercises described in *Section 5.2.1*, this section highlights a few other types of Arduino compatible sensors (<\$100) that could be added to LEGOLAS. One of these is an aqueous conductivity sensor (BNC compatible) that might be lowered into sample wells by a similar winch mechanism as for the pH sensor (*Figure 2.1*). Another is an arduino spectrophotometer attachment that could be used to conduct spectroscopy measurements in the 410 nm (UV) to 940 nm (IR) range. Lastly, a 12V, 80W Arduino compatible heating element (~220°C max) could be utilized with integrated temperature feedback to add temperature as a processing parameter in any studies (*Figure 5.2*).

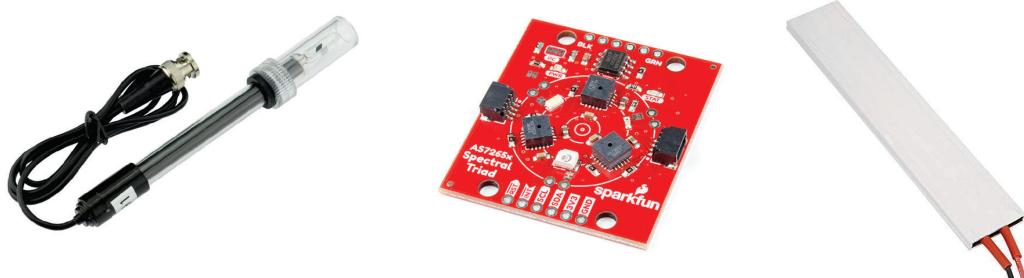


Figure 5.2: (left) Analog aqueous conductivity probe [102], (middle) spectrophotometer [103], and (right) heating element [104].

5.2 Conclusions

I developed LEGOLAS as a low-cost educational SDL and implemented it in the undergraduate and graduate classroom environments. It was shown to help highlight deficiencies in AE knowledge, encourage critical thinking with regards to pros and cons of AE, and help build upon teamwork, programming, hardware proficiency, and data management skills. As a hands-on tool, it was valuable in captivating student interest and providing a transferable example to real-world AE problems. The modular nature of the robot also facilitates a potential for the adoption of alternative studies in the future. I also used LEGOLAS to run closed-loop live runs with several active learning schemes: Bayesian Optimization, Bayesian Inference, and hypothesis generation and validation. The low-cost nature of LEGOLAS and its ability to serve as a high quality educational tool are promising in ensuring it has a large educational impact. In the coming years, I hope that LEGOLAS or other similar tools are implemented in applied ML for science courses to encourage young students to pursue and improve the burgeoning field of AE.

Appendices

I: Associated Media

Videos:

“Autonomous Liquid Mixing Robot - Active Learning” [105]

- *LEGOLAS (G-I) using a Gaussian Process for an exploration based campaign objective*

“LEGOLAS System” [86]

- *LEGOLAS (G-III) demonstrating the hypothesis validation active learning schema*

Publications:

- LEGOLAS Paper (2022) [24]
- MRS Presentation [23]
- NIST SURF Presentation [22]

Workshops:

- ALPha Workshop (2023) [106]
- Machine Learning Summer Bootcamp (2022) [107]

Bibliography

- [1] Xu, M., David, J. M., & Kim, S. H. (2018). The Fourth Industrial Revolution: Opportunities and Challenges. *International Journal of Financial Research*, 9(2), 90-95.
<https://doi.org/10.5430/ijfr.v9n2p90>
- [2] Groumpos, P. P. (2021). A critical historical and scientific overview of all industrial revolutions. *IFAC-PapersOnLine*, 54(13), 464-471.
<https://doi.org/10.1016/j.ifacol.2021.10.492>
- [3] Helfgott, R. B. (1986). America's third Industrial Revolution. *Challenge*, 29(5), 41-46.
<https://doi.org/10.1080/05775132.1986.11471116>
- [4] Potyrailo, R., Rajan, K., Stoewe, K., Takeuchi, I., Chisholm, B., & Lam, H. (2011). Combinatorial and high-throughput screening of materials libraries: Review of state of the art. *ACS Combinatorial Science*, 13(6), 579-633. <https://doi.org/10.1021/co200007w>
- [5] Anderson, P. W. (1972). More is different. *Science*, 177(4047), 393-396.
<https://doi.org/10.1126/science.177.4047.393>
- [6] Anderson, P. W., & Abrahams, E. (2009). Machines fall short of revolutionary science. *Science*, 324(5934), 1515-1516. https://doi.org/10.1126/science.324_1515c
- [7] Tabor, D. P., Roch, L. M., Saikin, S. K., Kreisbeck, C., Sheberla, D., Montoya, J. H., Dwaraknath, S., Aykol, M., Ortiz, C., Tribukait, H., Amador-Bedolla, C., Brabec, C. J., Maruyama, B., Persson, K. A., & Aspuru-Guzik, A. (2018). Accelerating the discovery of materials for clean energy in the era of smart automation. *Nature Reviews Materials*, 3(5), 5-20. <https://doi.org/10.1038/s41578-018-0005-z>
- [8] Kulik, H. J., & Tiwary, P. (2022). Artificial intelligence in computational materials science. *MRS Bulletin*, 47, 927-929.
- [9] Carbone, M. R. (2022). When not to use machine learning: A perspective on potential and limitations. *MRS Bulletin*, 47(9), 968-974. <https://doi.org/10.1557/s43577-022-00417-z>
- [10] Schmidt, E. (2023, March 2). Innovation power: Why Technology Will Define the Future of Geopolitics. *Foreign Affairs*.
<https://www.foreignaffairs.com/united-states/eric-schmidt-innovation-power-technology-geopolitics>
- [11] Johnson, A. (2023, March 30). Which Jobs will AI Replace? *Forbes*.
<https://www.forbes.com/sites/ariannajohnson/2023/03/30/which-jobs-will-ai-replace-the-4-industries-will-be-heavily-impacted/?sh=ac04a8b5957f>

- [12] Hippalgaonkar, K., Li, Q., Wang, X., Fisher, J. W., Kirkpatrick, J., & Buonassisi, T. (2023). Knowledge-integrated machine learning for materials: Lessons from gameplaying and robotics. *Nature Reviews Materials*. <https://doi.org/10.1038/s41578-022-00513-1>
- [13] MacLeod, B. P., Parlante, F. G., Brown, A. K., Hein, J. E., & Berlinguette, C. P. (2021). Flexible automation accelerates materials discovery. *Nature Materials*, 21(7), 722-726. <https://doi.org/10.1038/s41563-021-01156-3>
- [14] Kalinin, S. V., Ziatdinov, M., Ahmadi, M., Ghosh, A., Roccapirore, K., Liu, Y., & Vasudevan, R. K. (2023). Designing Workflows for Materials Characterization. *arXiv*. <https://arxiv.org/abs/2302.04397>
- [15] Friederich, P., Krenn, M., Tamblyn, I., & Aspuru-Guzik, A. (2021). Scientific intuition inspired by machine learning-generated hypotheses. *Machine Learning: Science and Technology*, 2(2), 025027. <https://doi.org/10.1088/2632-2153/abda08>
- [16] Xie, Y., Sattari, K., Zhang, C., & Lin, J. (2023). Toward autonomous laboratories: Convergence of artificial intelligence and experimental automation. *Progress in Materials Science*, 132, 101043. <https://doi.org/10.1016/j.pmatsci.2022.101043>
- [17] King, R. D. (2009). The Automation of Science. *Science*, 324(5923), 85-89. <https://doi.org/10.1126/science.1165620>
- [18] Abolhasani, M., & Kumacheva, E. (2023). The rise of self-driving labs in chemical and materials sciences. *Nature Synthesis*. <https://doi.org/10.1038/s44160-022-00231-0>
- [19] Ritchie, A. (2020). The impact of creating the next-generation materials genome initiative workforce. *JOM*, 72(7), 2466-2468. <https://doi.org/10.1007/s11837-020-04230-4>
- [20] Reynolds, E. B., & Uygun, Y. (2018). Strengthening advanced manufacturing innovation ecosystems: The case of Massachusetts. *Technological Forecasting and Social Change*, 136, 178-191. <https://doi.org/10.1016/j.techfore.2017.06.003>
- [21] Benešová, A., & Tupa, J. (2017). Requirements for education and qualification of people in industry 4.0. *Procedia Manufacturing*, 11, 2195-2202. <https://doi.org/10.1016/j.promfg.2017.07.366>
- [22] Saar, L. M. (2021, August). Bayesian Active Learning for Autonomous Physical Science. *NIST Summer Undergraduate Research Fellowship Colloquium [Symposium]*. .
- [23] Saar, L. (2021, December). A Low-Cost Education Platform for Teaching Autonomous Physical Science. *Symposium on Accelerating Experimental Materials Research with Machine Learning [Symposium]*. *Materials Research Society*, Boston. https://github.com/logansaar/LEGOLAS-Files/blob/main/Presentations/Saar_MRS_Talk.pdf

- [24] Saar, L., Liang, H., Wang, A., McDannald, A., Rodriguez, E., Takeuchi, I., & Kusne, A. G. (2022). The LEGOLAS kit: A low-cost robot science kit for education with symbolic regression for hypothesis discovery and validation. *MRS Bulletin*, 47(9), 881-885. <https://doi.org/10.1557/s43577-022-00430-2>
- [25] Schmidt, M., & Lipson, H. (2009). Distilling Free-Form Natural Laws from Experimental Data. *Science*, 324(5923), 81-85. DOI: 10.1126/science.1165893
- [26] Shannon, C. E., & Weaver, W. (1949). The mathematical theory of communication.
- [27] Stach, E., DeCost, B., Kusne, A. G., Hattrick-Simpers, J., Brown, K. A., Reyes, K. G., Schrier, J., Billinge, S., Buonassisi, T., Foster, I., Gomes, C. P., Gregoire, J. M., Mehta, A., Montoya, J., Olivetti, E., Park, C., Rotenberg, E., Saikin, S. K., Smullin, S., ... Maruyama, B. (2021). Autonomous experimentation systems for materials development: A community perspective. *Matter*, 4(9), 2702-2726. <https://doi.org/10.1016/j.matt.2021.06.036>
- [28] McDannald, A., Frontzek, M., Savici, A. T., Doucet, M., Rodriguez, E. E., Meuse, K., Opsahl-Ong, J., Samarov, D., Takeuchi, I., Ratcliff, W., & Kusne, A. G. (2022). On-the-fly autonomous control of neutron diffraction via physics-informed Bayesian active learning. *Applied Physics Reviews*, 9(2), 021408. <https://doi.org/10.1063/5.0082956>
- [29] Wagner, J., Berger, C. G., Du, X., Stubhan, T., Hauch, J. A., & Brabec, C. J. (2021). The evolution of materials acceleration platforms: Toward the laboratory of the future with AMANDA. *Journal of Materials Science*, 56(29), 16422-16446. <https://doi.org/10.1007/s10853-021-06281-7>
- [30] Christensen, M., Yunker, L. P., Shiri, P., Zepel, T., Prieto, P. L., Grunert, S., Bork, F., & Hein, J. E. (2021). Automation isn't automatic. *Chemical Science*, 12(47), 15473-15490. <https://doi.org/10.1039/d1sc04588a>
- [31] Baker, M. (2016). 1,500 scientists lift the lid on reproducibility. *Nature*, 533, 452-454. <https://doi.org/10.1038/533452a>
- [32] Gongora, A. E., Snapp, K. L., Whiting, E., Riley, P., Reyes, K. G., Morgan, E. F., & Brown, K. A. (2020). Using simulation to accelerate autonomous experimentation (AE): A case study using mechanics. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.3751791>
- [33] Mead, R. (1990). The design of experiments: Statistical principles for practical applications. Cambridge University Press
- [34] Kusne, A. G., Yu, H., Wu, C., Zhang, H., Hattrick-Simpers, J., DeCost, B., Sarker, S., Oses, C., Toher, C., Curtarolo, S., Davydov, A. V., Agarwal, R., Bendersky, L. A., Li, M.,

- Mehta, A., & Takeuchi, I. (2020). On-the-fly closed-loop materials discovery via Bayesian active learning. *Nature Communications*, 11(1).
<https://doi.org/10.1038/s41467-020-19597-w>
- [35] Tourlomousis, F. (n.d.). About superlabs. superlabs.eu. <https://superlabs.eu/about-us/>
- [36] Liang, Q., Gongora, A. E., Ren, Z., Tiihonen, A., Liu, Z., Sun, S., Deneault, J. R., Bash, D., Mekki-Berrada, F., Khan, S. A., Hippalgaonkar, K., Maruyama, B., Brown, K. A., Fisher III, J., & Buonassisi, T. (2021). Benchmarking the performance of Bayesian optimization across multiple experimental materials science domains. *npj Computational Materials*, 7(1). <https://doi.org/10.1038/s41524-021-00656-9>
- [37] Görtler, J., Kehlbeck, R., & Deussen, O. (2019, April 2). A visual exploration of gaussian processes. *Distill*. <https://doi.org/10.23915/distill.00017>
- [38] NSS. (2023, February 21). Power of Bayesian Statistics & Probability | Data Analysis. *Analytics Vidhya*.
<https://www.analyticsvidhya.com/blog/2016/06/bayesian-statistics-beginners-simple-english/>
- [39] Bongard, J., Schmidt, M., & Lipson, H. (2009). Eureqa. *Creative Machines Lab - Columbia University*. <https://www.creativemachineslab.com/eureqa.html>
- [40] Nikolaev, P., Hooper, D., Webber, F., Rao, R., Decker, K., Krein, M., Poleski, J., Barto, R., & Maruyama, B. (2016). Autonomy in materials research: A case study in carbon nanotube growth. *npj Computational Materials*, 2(1).
<https://doi.org/10.1038/npjcompumats.2016.31>
- [41] Chang, J., Nikolaev, P., Carpén-Núñez, J., Rao, R., Decker, K., Islam, A. E., Kim, J., Pitt, M. A., Myung, J. I., & Maruyama, B. (2020). Efficient closed-loop maximization of carbon nanotube growth rate using Bayesian optimization. *Scientific Reports*, 10(1).
<https://doi.org/10.1038/s41598-020-64397-3>
- [42] Gongora, A. E., Xu, B., Perry, W., Okoye, C., Riley, P., Reyes, K. G., Morgan, E. F., & Brown, K. A. (2020). A Bayesian experimental autonomous researcher for mechanical design. *Science Advances*, 6(15). <https://doi.org/10.1126/sciadv.aaz1708>
- [43] Ren, F., Ward, L., Williams, T., Laws, K. J., Wolverton, C., Hattrick-Simpers, J., & Mehta, A. (2018). Accelerated discovery of metallic glasses through iteration of machine learning and high-throughput experiments. *Science Advances*, 4(4).
<https://doi.org/10.1126/sciadv.aaq1566>
- [44] Zhang, L., Lin, D., Wang, H., Car, R., & E, W. (2019). Active learning of uniformly accurate interatomic potentials for materials simulation. *Physical Review Materials*, 3(2).
<https://doi.org/10.1103/physrevmaterials.3.023804>

- [45] Bash, D., Chenardy, F. H., Ren, Z., Cheng, J. J., Buonassisi, T., Oliveira, R., Kumar, J. N., & Hippalgaonkar, K. (2022). Accelerated automated screening of viscous graphene suspensions with various surfactants for optimal electrical conductivity. *Digital Discovery*, 1(2), 139-146. <https://doi.org/10.1039/d1dd00008j>
- [46] Masubuchi, S., Morimoto, M., Morikawa, S., Onodera, M., Asakawa, Y., Watanabe, K., Taniguchi, T., & Machida, T. (2018). Autonomous robotic searching and assembly of two-dimensional crystals to build van Der waals superlattices. *Nature Communications*, 9(1). <https://doi.org/10.1038/s41467-018-03723-w>
- [47] Macleod, B. P., Parlame, F. G., Morrissey, T. D., Häse, F., Roch, L. M., Dettelbach, K. E., & Moreira, R. (2020). Self-driving laboratory for accelerated discovery of thin-film materials. *Science Advances*, 6(20). <https://doi.org/10.1126/sciadv.aaz886>
- [48] Li, J., Li, J., Liu, R., Tu, Y., Li, Y., Cheng, J., He, T., & Zhu, X. (2020). Autonomous discovery of optically active chiral inorganic perovskite nanocrystals through an intelligent cloud lab. *Nature Communications*, 11(1). <https://doi.org/10.1038/s41467-020-15728-5>
- [49] Seifrid, M., Pollice, R., Aguilar-Granda, A., Morgan Chan, Z., Hotta, K., Ser, C. T., Vestfrid, J., Wu, T. C., & Aspuru-Guzik, A. (2022). Autonomous chemical experiments: Challenges and perspectives on establishing a self-driving lab. *Accounts of Chemical Research*, 55(17), 2454-2466. <https://doi.org/10.1021/acs.accounts.2c00220>
- [50] Burger, B., Maffettone, P. M., Gusev, V. V., Aitchison, C. M., Bai, Y., Wang, X., Li, X., Alston, B. M., Li, B., Clowes, R., Rankin, N., Harris, B., Sprick, R. S., & Cooper, A. I. (2020). A mobile robotic chemist. *Nature*, 583(7815), 237-241. <https://doi.org/10.1038/s41586-020-2442-2>
- [51] Coley, C. W., Thomas, D. A., Lummiss, J. A., Jaworski, J. N., Breen, C. P., Schultz, V., Hart, T., Fishman, J. S., Rogers, L., Gao, H., Hicklin, R. W., Plehiers, P. P., Byington, J., Piotti, J. S., Green, W. H., Hart, A. J., Jamison, T. F., & Jensen, K. F. (2019). A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science*, 365(6453). <https://doi.org/10.1126/science.aax1566>
- [52] Steiner, S., Wolf, J., Glatzel, S., Andreou, A., Granda, J. M., Keenan, G., Hinkley, T., Aragon-Camarasa, G., Kitson, P. J., Angelone, D., & Cronin, L. (2019). Organic synthesis in a modular robotic system driven by a chemical programming language. *Science*, 363(6423). <https://doi.org/10.1126/science.aav2211>
- [53] Fully configurable solutions for lab automation. (n.d.). Chemspeed Technologies. <https://www.chemspeed.com/configurable-solutions/>
- [54] IBM RXN for chemistry. (n.d.). IBM RXN for Chemistry. <https://rxn.app.accelerate.science/rxn/robo-rxn/welcome>

- [55] Materials Genome Initiative Strategic Plan. (2021). *National Science and Technology Council, Committee on Technology, Subcommittee on the Materials Genome Initiative*. <https://www.mgi.gov/sites/default/files/documents/MGI-2021-Strategic-Plan.pdf>
- [56] Curriculum – data-enabled discovery and design of energy materials (D3EM). (n.d.). *Data-Enabled Discovery and Design of Energy Materials (D3EM) – Texas A&M University*. <https://d3em.tamu.edu/curriculum/>
- [57] Department of Materials Design and Innovation: Our Research. (2022, April 6). *School of Engineering and Applied Sciences - University at Buffalo*. <https://engineering.buffalo.edu/materials-design-innovation/research.html>
- [58] ASM materials camp®-teachers. (2022, December 1). *ASM Materials Education Foundation*. <https://www.asmfoundation.org/teachers/materials-camps/year-one/>
- [59] Data and Informatics Graduate Intern-Traineeship: Materials at the Atomic Scale. (n.d.). *National Center for Supercomputing Applications*. <https://digi-mat.ncsa.illinois.edu/>
- [60] AI for Understanding and Designing Materials (aiM). (2022, December 13). *Welcome to the aiM Program: a National Research Traineeship Program*. <https://aim-nrt.pratt.duke.edu/>
- [61] Summer School for Integrated Computational Materials Education. (n.d.). *Michigan Engineering*. <https://icmed.engin.umich.edu/>
- [62] Betz, J., Zheng, H., Zang, Z., Sauerbeck, F., Zheng, Y. R., Biswas, J., Walas, K., Dimitrov, V., Behl, M., Krovi, V., & Mangharam, R. (2022). Teaching Autonomous Systems Hands-On: Leveraging Modular Small-Scale Hardware in the Robotics Classroom. *arXiv*. arXiv:2209.11181v1
- [63] Autonomous vehicle challenge. (n.d.). *Rose-Hulman Institute of Technology*. <https://www.rose-hulman.edu/academics/educational-outreach/autonomous-vehicle-challenge/index.html>
- [64] Brand, I., Roy, J., Ray, A., Oberlin, J., & Oberlix, S. (2018). PiDrone: An autonomous educational drone using raspberry Pi and Python. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. <https://doi.org/10.1109/iros.2018.8593943>
- [65] Howell, A., Way, E., McGrann, R., & Woods, R. (2006). Autonomous robots as a generic teaching tool. *Proceedings. Frontiers in Education. 36th Annual Conference*. <https://doi.org/10.1109/fie.2006.322654>
- [66] Nourbakhsh, I. R., Crowley, K., Bhate, A., Hamner, E., Hsiu, T., Perez-Bergquist, A., Richards, S., & Wilkinson, K. (2005). The robotic autonomy mobile robotics course:

Robot design, curriculum design and educational assessment. *Autonomous Robots*, 18(1), 103-127. <https://doi.org/10.1023/b:auro.0000047303.20624.02>

- [67] Opentrons. (2022, August 25). *Opentrons.com*. <https://opentrons.com/>
- [68] Baird, S. G., & Sparks, T. D. (2022). What is a minimal working example for a self-driving laboratory? *Matter*, 5(12), 4170-4178. <https://doi.org/10.1016/j.matt.2022.11.007>
- [69] Caramelli, D., Salley, D., Henson, A., Camarasa, G. A., Sharabi, S., Keenan, G., & Cronin, L. (2018). Networking chemical robots for reaction multitasking. *Nature Communications*, 9(1). <https://doi.org/10.1038/s41467-018-05828-8>
- [70] Roch, L. M., Häse, F., & Aspuru-Guzik, A. (2020). Chapter 16. ChemOS: An orchestration software to democratize autonomous discovery. *Artificial Intelligence in Drug Discovery*, 349-388. <https://doi.org/10.1039/9781788016841-00349>
- [71] About SMART. (n.d.). *Singapore-MIT*. <https://smart.mit.edu/about-smart/about-smart>
- [72] Deneault, J. R., Chang, J., Myung, J., Hooper, D., Armstrong, A., Pitt, M., & Maruyama, B. (2021). Toward autonomous additive manufacturing: Bayesian optimization on a 3D printer. *MRS Bulletin*, 46(7), 566-575. <https://doi.org/10.1557/s43577-021-00051-1>
- [73] Jiang, Y. (2014, May 25). Build your very own drink mixing robot. *Yu Jiang Tham – Dream it. Build it.* <https://yujiangtham.com/2014/05/25/build-your-very-own-drink-mixing-robot-part-1/>
- [74] Materiom. (n.d.). *Materiom*. <https://materiom.org/>
- [75] Tourlomousis, F. (n.d.). About superlabs. superlabs.eu/about-us/
- [76] Yang, F., Lai, V., Legard, K., Kozdras, S., Prieto, P. L., Grunert, S., & Hein, J. E. (2021). Augmented titration setup for future teaching laboratories. *Journal of Chemical Education*, 98(3), 876-881. <https://doi.org/10.1021/acs.jchemed.0c01394>
- [77] Atinary SDLabs Demo. (2022, June 17). *YouTube*. <https://www.youtube.com/watch?v=SX26XRFx0U0>
- [78] Pomberger, A., Jose, N., Walz, D., Meissner, J., Holtze, C., Kopczynski, M., Müller-Bischof, P., & Lapkin, A. (2022). Automated pH adjustment driven by robotic workflows and active machine learning. *ChemRxiv*. <https://doi.org/10.26434/chemrxiv-2022-zrm6p>
- [79] Gutierrez, J. M., Hinkley, T., Taylor, J. W., Yanev, K., & Cronin, L. (2014). Evolution of oil droplets in a chemorobotic platform. *Nature Communications*, 5(1). <https://doi.org/10.1038/ncomms6571>

- [80] Arnold, C. (2022, June 13). Cloud labs: Where robots do the research. *Nature*.
<https://www.nature.com/articles/d41586-022-01618-x>
- [81] Li, J., Li, J., Liu, R., Tu, Y., Li, Y., Cheng, J., He, T., & Zhu, X. (2020). Autonomous discovery of optically active chiral inorganic perovskite nanocrystals through an intelligent cloud lab. *Nature Communications*, 11(1).
<https://doi.org/10.1038/s41467-020-15728-5>
- [82] Baas, S., & Saggiomo, V. (2021). Ender3 3D printer kit transformed into open, programmable syringe pump set. *HardwareX*, 10, 1-24.
<https://doi.org/10.1016/j.ohx.2021.e00219>
- [83] Arducam 5mp wide angle usb camera for computer, 1/4" Cmos ov5648 mini uvc usb2.0 video webcam without microphone, with 3.3ft/1m cable : Electronics. (n.d.).
Amazon.com..
https://www.amazon.com/Arducam-Camera-Computer-Without-Microphone/dp/B0972K97BC/?_encoding=UTF8&pd_rd_w=vIULH&content-id=amzn1.sym.08900ed4-dd64-49b1-bce7-2e717defb1aa&pf_rd_p=08900ed4-dd64-49b1-bce7-2e717defb1aa&pf_rd_r=MXZ0J70WS4KBTEP87G1Z&pd_rd_wg=XNWze&pd_rd_r=19a4751e-9f81-4e8a-b186-a34db1f3f36c&ref_=pd_gw_ci_mcx_mi
- [84] Gravity: Analog pH sensor - Meter kit. (n.d.). *Arduino Official Store*.
<https://store.arduino.cc/products/gravity-analog-ph-sensor-meter-kit>
- [85,6] Gerber, L. C., Calasanz-Kaiser, A., Hyman, L., Voitiuk, K., Patil, U., & Riedel-Kruse, I. H. (2017). Liquid-handling Lego robots and experiments for STEM education and research. *PLOS Biology*, 15(3), e2001413. <https://doi.org/10.1371/journal.pbio.2001413>
- [86] Saar, L. (2022, March 18). LEGOLAS System [Video]. *YouTube*.
<https://www.youtube.com/watch?v=TtPM7zXI5kQ&feature=youtu.be>
- [87] Malolo. (2019, July 1). Screw-less / snap fit customizable raspberry Pi 4 case & stands by Malolo. *Thingiverse - Digital Designs for Physical Objects*.
<https://www.thingiverse.com/thing:3723561>
- [88] De Levie, R. (2003). The Henderson-hasselbalch equation: Its history and limitations. *Journal of Chemical Education*, 80(2), 146. <https://doi.org/10.1021/ed080p146>
- [89] Titration Curve of a Weak Acid and its pKa. (2023, February 18). *EasyBiologyClass*.
<https://www.easybiologyclass.com/titration-curve-of-a-weak-acid-and-its-pka-biochemistry-notes/>
- [90] Carpentier, A., Lazaric, A., Ghavamzadeh, M., Munos, R., & Auer, P. (2011). Upper-confidence-Bound algorithms for active learning in multi-armed bandits. *Lecture Notes in Computer Science*, 189-203. https://doi.org/10.1007/978-3-642-24412-4_17

- [91] Feynman, R. (1964). Feynman on Scientific Method [Video]. *YouTube*.
<https://www.youtube.com/watch?v=EYPapE-3FRw>
- [92] Domingos, P. (1970). The Role of Occam's Razor in Knowledge Discovery. *Data Mining and Knowledge Discovery*, 3, 409-425. <https://doi.org/10.1023/A:1009868929893>
- [93] Bayesian Information Criterion. (n.d.). *instructional media + magic*.
<https://www.immagic.com/eLibrary/ARCHIVES/GENERAL/WIKIPEDI/W120607B.pdf>
- [94] Normal (Gaussian) Distribution. (n.d.). *Probability, Statistics & Random Processes*.
https://www.probabilitycourse.com/chapter4/4_2_3_normal.php
- [95] Hernández-Lobato, J. M., Hoffman, M. W., & Ghahramani, Z. (2014). Predictive Entropy Search for Efficient Global Optimization of Black-box Functions. *arXiv*.
<https://doi.org/10.48550/arXiv.1406.2541>
- [96] Hennig, P., & Schuler, C. J. (2012). Entropy Search for Information-Efficient Global Optimization. *Journal of Machine Learning Research*, 13, 1809-1837.
<https://doi.org/10.48550/arXiv.1112.1217>
- [97] Wang, Z., & Jegelka, S. (2017). Max-value Entropy Search for Efficient Bayesian Optimization. *arXiv*. <https://doi.org/10.48550/arXiv.1703.01968>
- [98] Takeno, S., Fukuoka, H., Tsukada, Y., Koyama, T., Shiga, M., Takeuchi, I., & Karasuyama, M. (2019). Multi-fidelity Bayesian Optimization with Max-value Entropy Search and its parallelization. *arXiv*. <https://doi.org/10.48550/arXiv.1901.08275>
- [99] Huber, M. F., Bailey, T., Durrant-Whyte, H., & Hanebeck, U. D. (2008). On entropy approximation for gaussian mixture random vectors. *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*.
<https://doi.org/10.1109/mfi.2008.4648062>
- [100] Iodine clock reaction. (n.d.). *Imagination Station*.
<https://www.imaginationstationtoledo.org/education-resources/diy-activities/iodine-clock-reaction>
- [101] Reaction Kinetics: The Iodine Clock Reaction. (n.d.). *Bellevue College, Washington*.
https://www.bellevuecollege.edu/wp-content/uploads/sites/140/2014/06/Kinetics_Iodine_Clock_Rxn_Reportssheets.pdf
- [102] Gravity: Analog electrical conductivity sensor / Meter for Arduino. (n.d.). *Arduino Online Shop*.
<https://store-usa.arduino.cc/collections/sensors-environment/products/gravity-analog-electrical-conductivity-sensor-meter-for-arduino>

- [103] SparkFun triad spectroscopy sensor - AS7265x (Qwiic). (n.d.). *SparkFun Electronics*.
https://www.sparkfun.com/products/15050?gclid=CjwKCAjwrJ-hBhB7EiwAuyBVXXPFX9qngR1xJiVYG4MWs6yn5ehuS9U864w2nNNbAbEXpAdX2SYjGBoCR8sQAvD_BwE#reviews
- [104] Arduino - Controls heating element | Arduino tutorial. (n.d.). *Arduino Getting Started*.
<https://arduinogetstarted.com/tutorials/arduino-controls-heating-element>
- [105] Saar, L. (2021, July 28). Autonomous Liquid Mixing Robot - Active Learning [Video].
YouTube. <https://www.youtube.com/watch?v=Ul9sx29vAXE&feature=youtu.be>
- [106] Kusne, A. G. (2022). LEGOLAS: LEGO-based Low-Cost Autonomous Scientist.
Experimental Physics | Advanced Laboratory Physics Association.
https://advlab.org/Imm2023Maryland_LEGOLAS
- [107] Machine learning for materials research 2022. (2022). University of Maryland - MSE.
<https://umd-mse.catalog.instructure.com/courses/mlmr-2022>