What is a minimal working example for a self-driving laboratory?

Sterling G. Baird[a,*], Taylor D. Sparks[a]

[a]*Department of Materials Science and Engineering, University of Utah, Salt Lake City, UT 84108, USA*

Abstract

Self-driving laboratories (SDLs) are the future; however, the capital and expertise required can be daunting. We introduce the idea of an optimization task for less than $100, a square foot of desk space, and an hour of total setup time from the shopping cart to the first "autonomous drive." We use optics rather than chemistry for our demo; after all, light is easier to move than matter. While not materials-based, importantly, several core principles of a self-driving materials discovery lab are retained in this cross-domain example: sending commands to hardware to adjust physical parameters, receiving measured objective properties, decision-making via active learning, and utilizing cloud-based simulations. The demo is accessible, extensible, modular, and repeatable, making it an ideal candidate for both low-cost prototyping of SDL concepts and learning principles of SDLs in a low-risk setting.

Keywords: materials acceleration platform, autonomous experimentation, lab of the future

1. Introduction

Data informatics applied to chemistry and materials science have led to many computationally and experimentally validated discoveries [1–3]. As the accessibility to robotics and advanced optimization algorithms has increased, there has been a shift towards implementing self-driving laboratories (SDLs) for materials discovery (i.e. materials acceleration platforms (MAPs)) [4–16]. These systems can be expensive and often require expertise across a range of disciplines. Several excellent platforms in chemistry and materials science for low-cost SDLs have been developed [17–22] which can serve as both educational and research tools. For wider adoption of a low-cost demo, the system needs to be cheaper, smaller, and simpler to set up while still preserving many functional aspects of a MAP.

In programming, a minimal working example (MWE) "is a code snippet that can be copied-and-pasted into an empty ... file and still have the same features (working) and that does not include unnecessary details (minimal). [23]" Here, we pose the question:

> What does a minimal working example look like for a self-driving laboratory?

To elaborate the connection, we provide our interpretation of corresponding definitions for a minimal, complete, reproducible programming example [24] applied to SDLs in Table 1.

In this work, we provide an overview of the self-driving laboratory demonstration (SDL-Demo) including required and optional bills of materials and hardware/software setup in Section 2. We then discuss limitations and design considerations (Section 3), extensions (Section 4), task complexity (Section 5), and hardware, software, and task alternatives (Section 6). Finally, we describe milestones, deliverables, and outlook for the project in Section 7.

*Corresponding author.

Email addresses: sterling.baird@utah.edu (Sterling G. Baird), sparks@eng.utah.edu (Taylor D. Sparks)

Table 1: Definitions of "minimal", "complete", and "reproducible" in the traditional programming context of minimal working examples and SDLs.

	Programming	Self-driving Laboratory
Minimal	Use as little code as possible that still produces the same problem [23]	Minimize the cost, size, and setup while still being a SDL
Complete	Provide all parts needed to reproduce the problem in the question itself [23]	Provide software with documentation and a bill of materials with setup instructions
Reproducible	Test the code you're about to provide to make sure it reproduces the problem [23]	Benchmark the SDL using a fixed configuration and verify the results are expected

2. Self-driving Laboratory Demo Overview

We introduce the idea of an optimization task for less than $100, a square foot of desk space, and an hour of total setup time. We believe our SDL-Demo adequately meets the minimal, complete, and reproducible requirements of a MWE SDL (Table 1) and meets the non-materials aspects of a MAP [13]:

> [A system that] carries out high-throughput and/or automated experiments, the results of which are fed back into the AI that guides the selection of subsequent rounds of experimentation to optimize or make a discovery.

The SDL-Demo involves controlling the brightness of a red green blue (RGB) light-emitting diode (LED), sensing the light mixture via a discrete-channel spectrophotometer, decision-making to tune the inputs to best match a desired spectrum, and optionally, cloud-based simulations to aid in decision-making. The setup is summarized in Figure 1 with required and optional bills of materials given in Figures 2 and 3, respectively.

The basic steps and substeps of assembling the hardware and running the demo—connecting components, mounting the sensor, setting up the RPi Pico W, and remote access—are given in Table 2.

3. Limitations, Design Considerations

Something unique to our approach is that there are no robotic movements in the default configuration. While this can be considered a limitation, we also consider it to be a strength because it dramatically reduces the cost, lessens the expertise required, and reduces the chance for initial closed-loop failure. There is still a need for a low-cost robotic MWE for SDLs which could serve as a complementary and more advanced extension to SDL-Demo (this work).

While the capital involved for this demo is low, it's possible that this could perpetuate the practice of only demonstrating rather than dedicating effort to materials acceleration for societal solutions (MASS) tasks [13]. In other words, the time that could potentially be spent modifying and benchmarking this setup can follow Boyle's law in expanding to fill the space available and siphon the "air" (resources) that might otherwise have been applied to a MASS task. In order to mitigate this risk, we recommend that researchers interested in extending the framework do so as a miniature testing piece and limited-scope stepping stone for a larger, established plan to create a MAP geared towards a MASS.

4. Extensions

As an illustrative example of using the SDL-Demo in the context of a larger plan, this demo could be extended to accommodate a distributed autonomous laboratory framework, where multiple copies of the demo are implemented at separate locations and operate collaboratively with

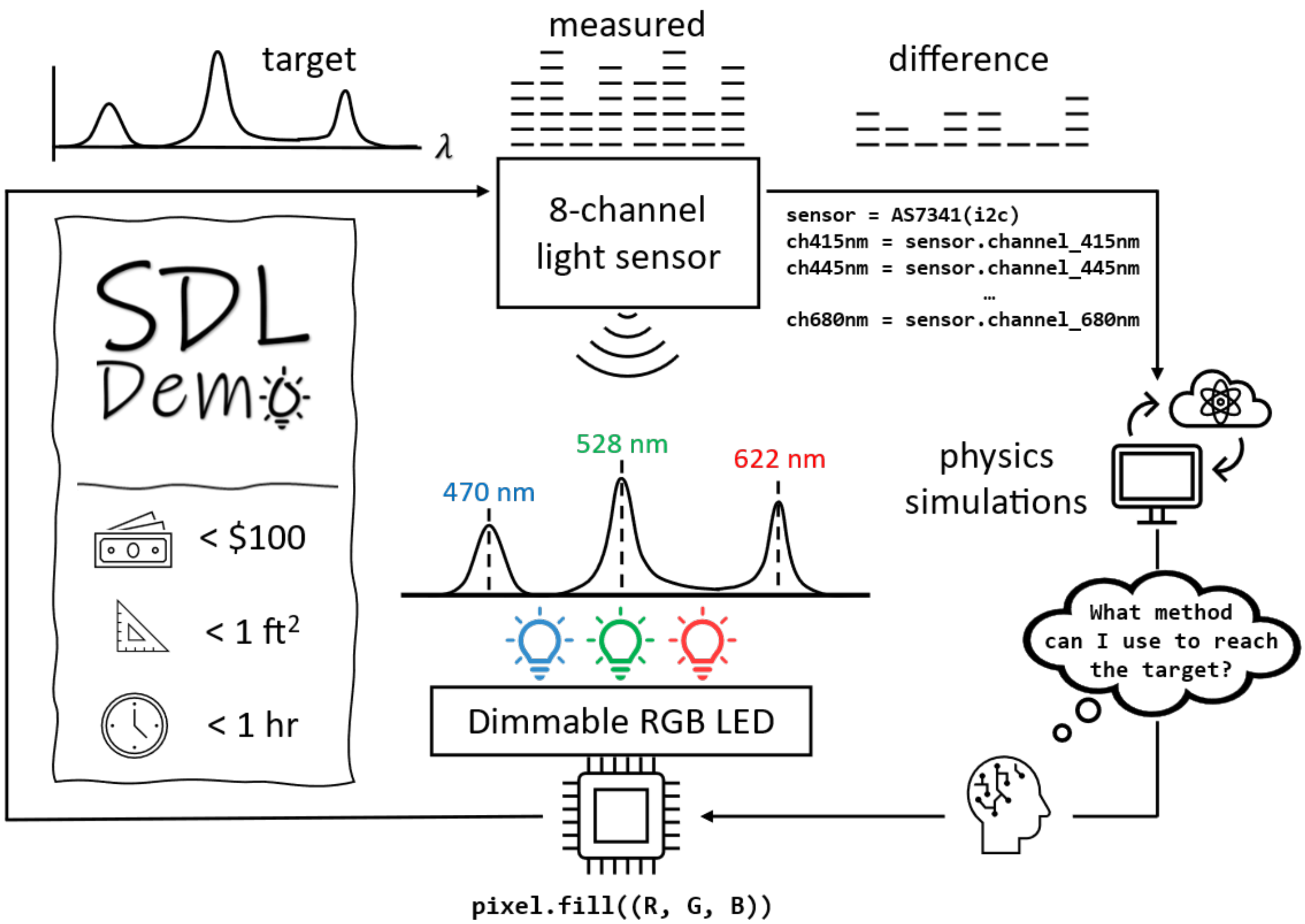
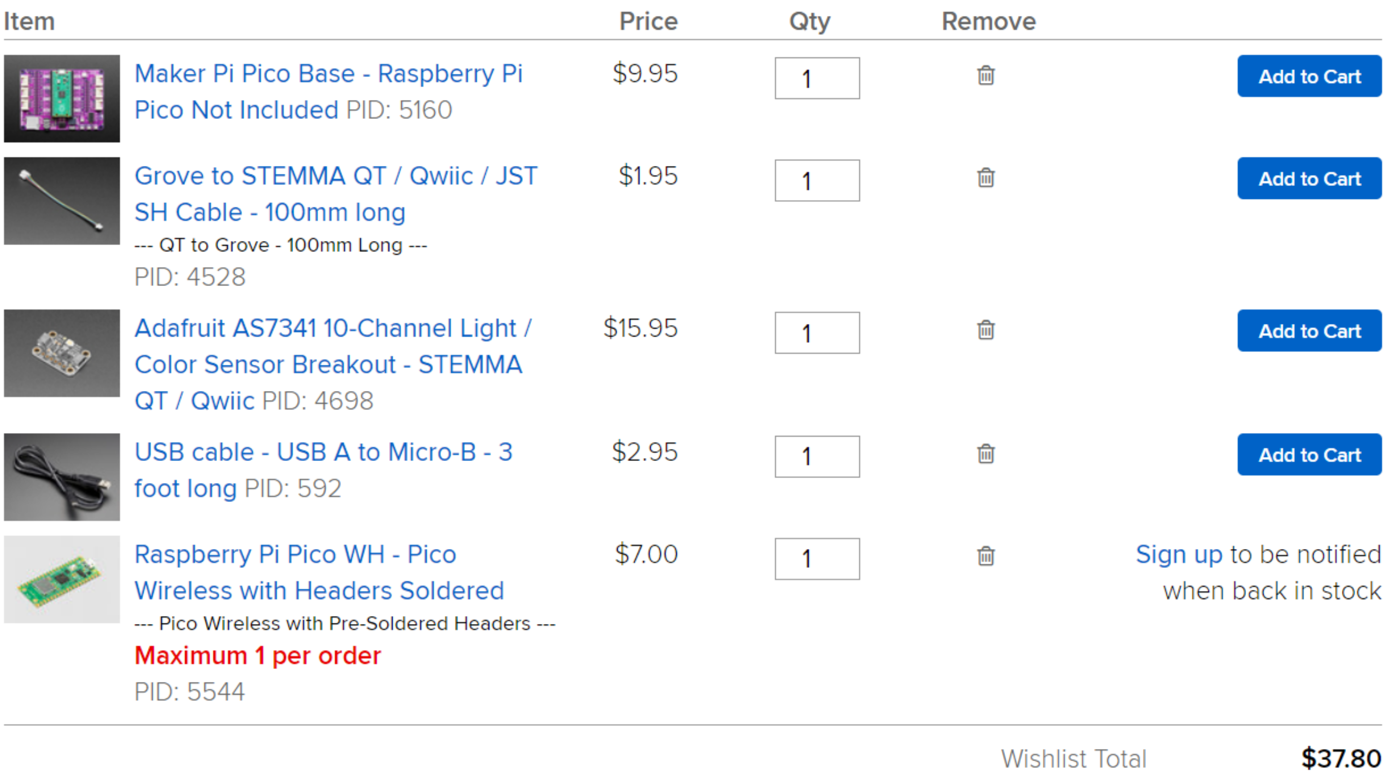
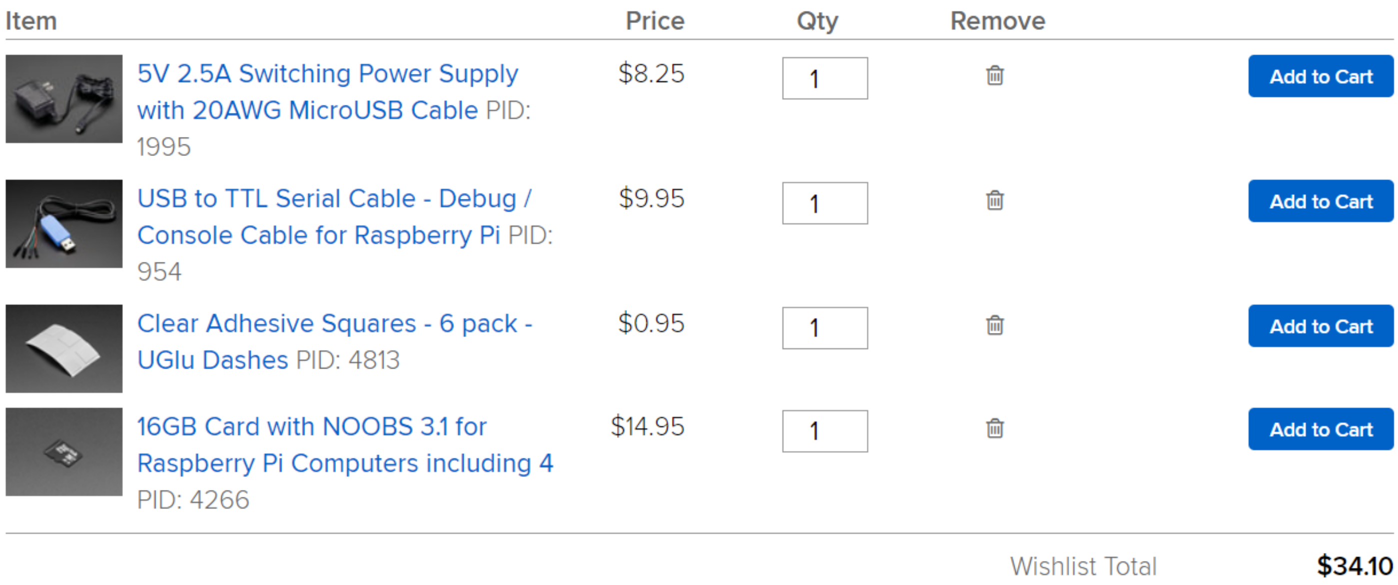


Figure 1: Summary of the self-driving laboratory demonstration (SDL-Demo). A microcontroller (Raspberry Pi (RPi)) sends commands to a dimmable red green blue (RGB) light-emitting diode (LED) to control the brightness at different wavelengths. A spectrophotometer measures the light signal at eight individual wavelengths. The microcontroller reads the intensity values from the spectrophotometer and uses these newly measured values and prior information (including e.g. prior measurements and physics-based simulations performed in the cloud) to choose the next set of LED parameters in an effort to better match a target spectrum. The setup adequately meets the minimal requirement of a minimal working example (MWE) self-driving laboratory (SDL) by costing less than 100 USD, occupying less than $1\,\text{ft}^2$ ($0.1\,\text{m}^2$) of desk space, and requiring less than 1 h of setup time.



Item	Price	Qty	Remove	
Maker Pi Pico Base - Raspberry Pi Pico Not Included PID: 5160	$9.95	1		Add to Cart
Grove to STEMMA QT / Qwiic / JST SH Cable - 100mm long --- QT to Grove - 100mm Long --- PID: 4528	$1.95	1		Add to Cart
Adafruit AS7341 10-Channel Light / Color Sensor Breakout - STEMMA QT / Qwiic PID: 4698	$15.95	1		Add to Cart
USB cable - USB A to Micro-B - 3 foot long PID: 592	$2.95	1		Add to Cart
Raspberry Pi Pico WH - Pico Wireless with Headers Soldered --- Pico Wireless with Pre-Soldered Headers --- **Maximum 1 per order** PID: 5544	$7.00	1		Sign up to be notified when back in stock
			Wishlist Total	**$37.80**

Figure 2: Bill of materials for required hardware to assemble the SDL-Demo using a Raspberry Pi (RPi) Pico WH. This Adafruit "wishlist" is available publicly at http://www.adafruit.com/wishlists/553992. This hardware configuration was designed to require no soldering and leverages Stemma-QT and Grove ports for easy interfacing between the RPi, Maker Pi Pico, and spectrophotometer. As an alternative to the Pico WH, a Pico W can be used, though it requires soldering the headers. Sculpting wire (14 American Wire Gauge) is recommended for adjustable mounting of the spectrophotometer relative to the Maker Pi Pico base RGB LED.



Item	Price	Qty	Remove	
5V 2.5A Switching Power Supply with 20AWG MicroUSB Cable PID: 1995	$8.25	1		Add to Cart
USB to TTL Serial Cable - Debug / Console Cable for Raspberry Pi PID: 954	$9.95	1		Add to Cart
Clear Adhesive Squares - 6 pack - UGlu Dashes PID: 4813	$0.95	1		Add to Cart
16GB Card with NOOBS 3.1 for Raspberry Pi Computers including 4 PID: 4266	$14.95	1		Add to Cart
			Wishlist Total	**$34.10**

Figure 3: Bill of materials for optional accessories for the SDL-Demo using a RPi Pico WH. This Adafruit "wishlist" is available publicly at `http://www.adafruit.com/wishlists/554001`. The optional hardware has three primary intentions: exposing additional general-purpose input/output pins for extending functionality of the demo, operating as a standalone computer package (i.e. no existing computer needed by adding a display, keyboard, and mouse), and providing an alternate method for setting up a "headless" RPi (i.e. when RPi must be accessed through a separate computer due to lack of standalone display, keyboard, and mouse).

Table 2: Hardware and software setup instructions for the SDL-Demo. Full instructions will be made available at `https://hackaday.io/project/186289-autonomous-research-laboratories`. In the interim, individual product pages from the bill of materials have links to hardware and software tutorials that will form the basis for the detailed SDL-Demo instructions.

Step	Substep
Connect components	Connect AS7341 to Maker Pi Pico base via Stemma-QT/Grove connector Insert RPi Pico W into Maker Pi Pico base
Mount the sensor	Thread sculpting wire through mounting holes on Maker Pi Pico base Thread same sculpting wire through mounting holes on AS7341 Position AS7341 perpendicular to and about 3 inches from NeoPixel LED
Set up Pico W	Hold BOOTSEL button, connect RPi to computer via micro-USB-B/USB-A Drag the latest Pico CircuitPython download onto the computer's `D:/` drive Install Thonny editor, configure for CircuitPython, and install libraries Replace `code.py` with the (web server) SDL-Demo version, click "Run"
Remote access	Install the SDL-Demo library to Google Colab or a local Python installation Remotely connect to the Pico W through the web server Run the basic SDL-Demo optimization script

model training and decision making happening in the cloud. This is another interesting aspect of the development of MAPs that could be explored in a low monetary risk setting; however, we believe this kind of extension to the demo would better serve as a proof-of-concept to be included in a grant proposal for a MASS MAP or as a test-bed for an existing distributed autonomous laboratory network working towards a MASS. See also the limited task complexity described in Section 5.

When used for education rather than research, we believe that similar considerations as mentioned above should be taken. In educational settings, equipment funding must be sourced. Successful implementation of the demo in classroom settings can provide a source of trust for more expensive, higher-impact demonstrations such as those involving movement of solids and liquids (robotics), changes in state variables (temperature, pressure), and multi-step syntheses. The SDL-Demo is a MWE that can help bolster confidence and motivate buy-in for future, larger scale implementations. This is similar to how developers are more likely to devote their time and resources to a programming question that contains a well thought out MWE.

The SDL-Demo can be used to explain what machine learning algorithms can be used for chemistry and materials science tasks and how they work. We are particularly interested in using SDL-Demo to convey important topics related to the efficiencies of various search algorithms: for example, a comparison of grid search vs. random search vs. Bayesian optimization. Optimization topics that are of interest to explore using the SDL-Demo are constrained [25, 26][1], multi-fidelity [27–32][2], and/or multi-objective [33–44] optimization.

The demo can also be *used directly* to prototype a system for a more advanced task. For example, the system could be converted from a light mixing demo to a chemical mixing demo by replacing the LED with an appropriate motor controller and peristaltic pump(s). For a chemistry-based color matching demo, the spectrophotometer could be used directly with longer integration times. Likewise, the light/sensor setup could be used to measure reflection, absorption, and transmission in various materials. For other tasks, the spectrophotometer could be replaced with the appropriate sensor (e.g. pH, temperature, conductivity). The ability to fall back to the original SDL-Demo also allows for more efficient, modular debugging and potentially less frustration for the user.

5. Task Complexity

There is nothing particularly complex about task of mixing several distinct wavelengths and matching a target spectrum; to a large extent, the spectrum response surface is linear with respect to the underlying inputs (red, green, and blue LED currents), aside from experimental noise. There exists only a single local optimum in the case of single-objective optimization of mean absolute error mismatch between measured and target spectra or a more robust metric such as Wasserstein distance between the discrete distributions. This can be contrasted with many chemistry and materials optimization tasks, where non-linear correlations, discontinuities, and multiple local optima come into play.

Depending on the use-case, the limited complexity of the SDL-Demo task can be seen as either a limitation or a strength. Used as a pedagogical tool, students are less likely to be overwhelmed. Used as a prototyping tool, debugging is likely to be more efficient and straightforward.

However, due to the relative simplicity, the SDL-Demo is of less interest from an optimization benchmarking scheme. Pulling again from a programming analogy, there is a phrase "duck typing" which refers to applying the duck test adage "If it walks like a duck and it quacks like a duck, then it must be a duck" to the concept of assigning types to variables (e.g. integers vs. floating-point). Adapted to the case of materials acceleration:

[1]The presence of search space degeneracies and how these are handled either explicitly or implicitly within an optimization algorithm are important for many, if not all materials optimization tasks.

[2]An example of discrete multi-fidelity optimization involves incorporating online and/or offline simulations into an experimental optimization scheme.

> If it looks like materials optimization and it behaves like materials optimization problem, then it must be a good benchmark for materials optimization.

Three input variables with linear responses neither looks like nor behaves like many materials optimization tasks; however, other benchmarking solutions exist. To this end, we are also developing a customizable computational benchmark as follow-up work to [45] that can be easily adapted to the number of constraints, input parameters, and outputs while retaining a more realistic response surface complexity.

While the SDL-Demo may be less suitable as a state-of-the-art benchmarking framework, we believe it can effectively serve as a hands-on teaching tool for optimization topics (Section 4) such as comparing search efficiency of well-known algorithms. Perhaps in future work, others may design a low-cost, self-driving experimental setup that retains input-output response complexity characteristic of many MASS tasks.

6. Alternatives

Because the design involves low-cost components that each come with pre-built Python libraries, the startup cost and time is minimal. While we propose a set of hardware and compatible software libraries, we comment on some alternatives here. For example, the Maker Pi Pico that contains an embedded RGB LED could be replaced by a custom printed circuit board with a single NeoPixel (or DotStar) RGB LED, a Blinkt! LEDs array, or a custom array of LEDs with many distinct wavelengths. In the cases involving custom printed circuit boards, an LED driver chip or board is likely necessary. An alternative to the AS7341 spectrophotometer is a do-it-yourself spectrophotometer; however, currently available open-source designs for spectrophotometers are likely to violate either the 100 USD cost or 1 hour setup time constraints outlined previously.

In the simplest setup, a single LED with a single brightness sensor could be used; however, this is missing qualitative features of SDLs for real-world tasks involving multiple tunable inputs and multiple signal measurements; it also presents additional hardware challenges and design considerations. For example, we wanted to keep the signal (i.e. LED) and sensor on separate boards while attached via a cable rather than integrating everything onto a single printed circuit board because it better mimics the SDL best practice of modularity [8, 11]. We did not find off-the-shelf components that adequately met these needs. While it would be possible to use a two-wire LED with a breadboard, breadboards can introduce insecure connections, a greater likelihood of wiring mistakes by novice users, and poor aesthetics. We argue that the first two issues impede the long-term extensibility of the SDL-Demo to other designs and applications while the latter issue of aesthetics may lead to less user appeal and lower adoption rates.

While this example is based on CircuitPython software, alternative computing languages such as MicroPython, Python, Arduino, and C/C++ are also viable, with preference towards languages with support for general-purpose input/output and ease-of-use. Rather than control the LEDs through CircuitPython libraries, a lower-level interface that directly controls electrical current could be employed. The use of other microcontrollers and single board computers are possible and would likely require only minor redesign for hardware peripherals and software.

We also note that while the use of LEDs seemed the most compatible with "Hello, World!" style electronics projects, alternative signals such as sound, Bluetooth, WiFi, and vibrational modes (e.g. a drumhead or water surface) could be used in a similar optimization scheme given the appropriate signal source and sensor hardware.

7. Milestones, Deliverables, and Outlook

Previously, we described limitations and design considerations (Section 3), extensions (Section 3), task complexity (Section 5), and alternatives (Section 6) in the context of SDL-Demo. Here, we describe basic milestones and deliverables for the project. Basic milestones involve ordering the bill of materials, assembling the system, setting up the microcontroller (i.e. RPi Pico W), unit testing individual components, writing the adaptive design script, and running the first "autonomous drive". There are four final deliverables: build instructions hosted on `https://hackaday.io` (project ID: 186289), software documentation and usage instructions hosted on GitHub (`https://github.com/sparks-baird/self-driving-lab-demo`), validation results, and a video demonstration/tutorial of an autonomous drive. It may also be worthwhile to package the system as a kit through a service such as Crowd Supply to accelerate buy-in and adoption and circumvent future supply chain problems.

Our goal is for every cheminformatics and materials informatics researcher or prospective student to have at least one hands-on exposure to implementing a SDL. We believe that as scientists, engineers, and educators implement this demonstration for prototyping and teaching the principles of SDLs at minimal cost, the community will get closer to the critical MASS [13] necessary for accelerating impactful materials discovery.

Glossary

LED light-emitting diode 2–4, 6, 7

MAP materials acceleration platform 1, 2, 6

MASS materials acceleration for societal solutions 2, 6–8

MWE minimal working example 1–3, 6

RGB red green blue 2–4, 7

RPi Raspberry Pi 2–5, 8

SDL self-driving laboratory 1–3, 7, 8

SDL-Demo self-driving laboratory demonstration 1–8

Conflicts of Interest

There are no conflicts of interest to declare.

Acknowledgement

We thank Ben Blaiszik for discussion and suggestions regarding a distributed autonomous laboratory. We thank Joshua Schrier for suggesting the use of RPi Pico over other RPi models and for feedback on the design philosophy. We thank Sergei Kalinin for suggestions related to DIY spectrophotometers. We thank Christoph Voelker for suggestions regarding evaluating performance of SDL-Demo. We thank Kjell Jorner, Tonio Buonassisi, Rhys Goodall, Logan Ward, and Alan Aspuru-Guzik for suggestions regarding search algorithms to use with the SDL-Demo. We thank Helen Leigh for suggesting the use of `https://hackaday.io/` for project tracking, and for suggestions related to hardware. We thank the support staff in the Adafruit forums for general feedback. This work was supported by the National Science Foundation under Grant No. DMR-1651668.

CRediT Statement

Sterling G. Baird: Conceptualization, Methodology, Investigation, Resources, Writing - Original Draft, Writing - Review & Editing, Visualization. **Taylor D. Sparks**: Supervision, Project administration, Funding acquisition

Data Availability

The open-source software and instructions related to this demo is being actively developed (as of 2022-07-09) at `https://github.com/sparks-baird/self-driving-lab-demo` and `https://hackaday.io/project/186289-autonomous-research-laboratories`. The open-source software for NeoPixel is available at `https://github.com/adafruit/Adafruit_CircuitPython_NeoPixel`. The open-source software for the AS7341 spectrophotometer is available at `https://github.com/adafruit/Adafruit_CircuitPython_AS7341`.

References

[1] S. G. Baird, M. Liu, H. M. Sayeed, T. D. Sparks, Data-driven materials discovery and synthesis using machine learning methods, in:

Comprehensive Inorganic Chemistry III, 2022. (in press).

[2] A. O. Oliynyk, J. M. Buriak, Virtual Issue on Machine-Learning Discoveries in Materials Science, Chem. Mater. 31 (2019) 8243–8247. doi:10.1021/acs.chemmater.9b03854.

[3] J. E. Saal, A. O. Oliynyk, B. Meredig, Machine Learning in Materials Discovery: Confirmed Predictions and Their Underlying Approaches, Annu. Rev. Mater. Res. 50 (2020) 49–69. doi:10.1146/annurev-matsci-090319-010954.

[4] C. W. Coley, N. S. Eyke, K. F. Jensen, Autonomous Discovery in the Chemical Sciences Part I: Progress, Angew. Chem. Int. Ed. 59 (2020) 22858–22893. doi:10.1002/anie.201909987.

[5] C. W. Coley, N. S. Eyke, K. F. Jensen, Autonomous Discovery in the Chemical Sciences Part II: Outlook, Angew. Chem. Int. Ed. 59 (2020) 23414–23436. doi:10.1002/anie.201909989.

[6] M. M. Flores-Leonar, L. M. Mejía-Mendoza, A. Aguilar-Granda, B. Sanchez-Lengeling, H. Tribukait, C. Amador-Bedolla, A. Aspuru-Guzik, Materials Acceleration Platforms: On the way to autonomous experimentation, Current Opinion in Green and Sustainable Chemistry 25 (2020) 100370. doi:10.1016/j.cogsc.2020.100370.

[7] B. Goldman, S. Kearnes, T. Kramer, P. Riley, W. P. Walters, Defining Levels of Automated Chemical Design, J. Med. Chem. 65 (2022) 7073–7087. doi:10.1021/acs.jmedchem.2c00334.

[8] B. P. MacLeod, F. G. L. Parlane, A. K. Brown, J. E. Hein, C. P. Berlinguette, Flexible automation accelerates materials discovery, Nat. Mater. (2021). doi:10.1038/s41563-021-01156-3.

[9] J. H. Montoya, M. Aykol, A. Anapolsky, C. B. Gopal, P. K. Herring, J. S. Hummelshøj, L. Hung, H.-K. Kwon, D. Schweigert, S. Sun, S. K. Suram, S. B. Torrisi, A. Trewartha, B. D. Storey, Toward autonomous materials research: Recent progress and future challenges, Applied Physics Reviews 9 (2022) 011405. doi:10.1063/5.0076324.

[10] J. M. Perkel, The Internet of Things comes to the lab, Nature 542 (2017) 125–126. doi:10.1038/542125a.

[11] F. Rahmanian, J. Flowers, D. Guevarra, M. Richter, M. Fichtner, J. Gregoire, H. S. Stein, Enabling Modular Autonomous Feedback-Loops in Materials Science through Hierarchical Experimental Laboratory Automation and Orchestration, Preprint, Chemistry, 2021. doi:10.33774/chemrxiv-2021-kr87t.

[12] M. Seifrid, R. Pollice, A. Aguilar-Granda, Z. Morgan Chan, K. Hotta, C. T. Ser, J. Vestfrid, T. C. Wu, A. Aspuru-Guzik, Autonomous Chemical Experiments: Challenges and Perspectives on Establishing a Self-Driving Lab, Acc. Chem. Res. (2022) acs.accounts.2c00220. doi:10.1021/acs.accounts.2c00220.

[13] M. Seifrid, J. Hattrick-Simpers, A. Aspuru-Guzik, T. Kalil, S. Cranford, Reaching critical MASS: Crowdsourcing designs for the next generation of materials acceleration platforms, Matter 5 (2022) 1972–1976. doi:10.1016/j.matt.2022.05.035.

[14] E. Stach, B. DeCost, A. G. Kusne, J. Hattrick-Simpers, K. A. Brown, K. G. Reyes, J. Schrier, S. Billinge, T. Buonassisi, I. Foster, C. P. Gomes, J. M. Gregoire, A. Mehta, J. Montoya, E. Olivetti, C. Park, E. Rotenberg, S. K. Saikin, S. Smullin, V. Stanev, B. Maruyama, Autonomous experimentation systems for materials development: A community perspective, Matter 4 (2021) 2702–2726. doi:10.1016/j.matt.2021.06.036.

[15] R. Vescovi, R. Chard, N. Saint, B. Blaiszik, J. Pruyne, T. Bicer, A. Lavens, Z. Liu, M. E. Papka, S. Narayanan, N. Schwarz, K. Chard,

I. Foster, Linking Scientific Instruments and HPC: Patterns, Technologies, Experiences, 2022. doi:10.48550/arXiv.2204.05128. arXiv:2204.05128.

[16] M. May, AUTOMATED SCIENCE ON A SHOESTRING, Nature 569 (2019) 587–588.

[17] D. Caramelli, D. Salley, A. Henson, G. A. Camarasa, S. Sharabi, G. Keenan, L. Cronin, Networking chemical robots for reaction multitasking, Nat Commun 9 (2018) 3406. doi:10.1038/s41467-018-05828-8.

[18] T. Fuhrmann, D. I. Ahmed, L. Arikson, M. Wirth, M. L. Miller, E. Li, A. Lam, P. Blikstein, I. Riedel-Kruse, Scientific Inquiry in Middle Schools by combining Computational Thinking, Wet Lab Experiments, and Liquid Handling Robots, in: Interaction Design and Children, ACM, Athens Greece, 2021, pp. 444–449. doi:10.1145/3459990.3465180.

[19] L. Saar, H. Liang, A. Wang, A. McDannald, E. Rodriguez, I. Takeuchi, A. G. Kusne, A Low-Cost Robot Science Kit for Education with Symbolic Regression for Hypothesis Discovery and Validation, 2022. doi:10.48550/arXiv.2204.04187. arXiv:2204.04187.

[20] Tonio Buonassisi [@toniobuonassisi], From Boston to Seattle, MIT students write machine-learning code to control an autonomous robot at SMART, MIT's research enterprise in Singapore. @LiuZhe_MIT queues up the next job for our color-matching robot, for our 2.s986 Applied ML class. #opentrons #appliedml #BoB @MITMechE https://t.co/LEnbzen6dc, 2020.

[21] J. M. P. Gutierrez, T. Hinkley, J. W. Taylor, K. Yanev, L. Cronin, Evolution of oil droplets in a chemorobotic platform, Nat Commun 5 (2014) 5571. doi:10.1038/ncomms6571.

[22] S. Vargas, S. Zamirpour, S. Menon, A. Rothman, F. Häse, T. Tamayo-Mendoza, J. Romero, S. Sim, T. Menke, A. Aspuru-Guzik, Team-Based Learning for Scientific Computing and Automated Experimentation: Visualization of Colored Reactions, J. Chem. Educ. 97 (2020) 689–694. doi:10.1021/acs.jchemed.9b00603.

[23] Lean Community, Minimal working examples, https://leanprover-community.github.io/mwe.html, 2022.

[24] Stack Overflow, How to create a Minimal, Reproducible Example - Help Center, https://stackoverflow.com/help/minimal-reproducible-example, 2022.

[25] S. G. Baird, J. R. Hall, T. D. Sparks, Effect of Reducible and Irreducible Search Space Representations on Adaptive Design Efficiency: A Case Study on Maximizing Packing Fraction for Solid Rocket Fuel Propellant Simulations, Preprint, Chemistry, 2022. doi:10.26434/chemrxiv-2022-nz2w8.

[26] R. J. Hickman, M. Aldeghi, F. Häse, A. Aspuru-Guzik, Bayesian optimization with known experimental and design constraints for chemistry applications, arXiv:2203.17241 [cond-mat] (2022). arXiv:2203.17241.

[27] J. M. Beaubien, The use of simulation for training teamwork skills in health care: How low can you go?, Quality and Safety in Health Care 13 (2004) i51–i56. doi:10.1136/qshc.2004.009845.

[28] K. T. Butler, F. Oviedo, P. Canepa, Machine Learning in Materials Science, American Chemical Society, Washington, DC, USA, 2022. doi:10.1021/acsinfocus.7e5033. arXiv:https://pubs.acs.org/doi/pdf/10.1021/ac

[29] C. Fare, P. Fenner, E. O. Pyzer-Knapp, A Principled Method for the Creation of Synthetic Multi-fidelity Data Sets, 2022. arXiv:2208.05667.

[30] S. Gong, S. Wang, T. Xie, W. H. Chae, R. Liu, J. C. Grossman, Calibrating DFT formation enthalpy calculations by multi-fidelity machine learning, 2022. doi:10.48550/arXiv.2110.13308. arXiv:2110.13308.

[31] B. Letham, E. Bakshy, Bayesian Optimization for Policy Search via Online-Offline Experimentation, arXiv:1904.01049 [cs, stat] (2019). arXiv:1904.01049.

[32] K. Kandasamy, K. R. Vysyaraju, W. Neiswanger, B. Paria, C. R. Collins, J. Schneider, B. Poczos, E. P. Xing, Tuning Hyperparameters without Grad Students: Scalable and Robust Bayesian Optimisation with Dragonfly, arXiv:1903.06694 [cs, stat] (2020). arXiv:1903.06694.

[33] Z. del Rosario, M. Rupp, Y. Kim, E. Antono, J. Ling, Assessing the frontier: Active learning, model accuracy, and multi-objective candidate discovery and optimization, J. Chem. Phys. 153 (2020) 024112. doi:10.1063/5.0006124.

[34] A. R. Falkowski, S. K. Kauwe, T. D. Sparks, Optimizing Fractional Compositions to Achieve Extraordinary Properties, Integrating Materials and Manufacturing Innovation (2021). doi:10.1007/s40192-021-00242-3.

[35] K. Hanaoka, Comparison of Conceptually Different Multi-Objective Bayesian Optimization Methods for Material Design Problems, Materials Today Communications (2022) 103440. doi:10.1016/j.mtcomm.2022.103440.

[36] K. Hanaoka, Comparison of Conceptually Different Multi-Objective Bayesian Optimization Methods for Material Design Problems, Materials Today Communications (2022) 103440. doi:10.1016/j.mtcomm.2022.103440.

[37] F. Häse, L. M. Roch, A. Aspuru-Guzik, Chimera: Enabling hierarchy based multi-objective optimization for self-driving laboratories, Chem. Sci. 9 (2018) 7642–7655. doi:10.1039/C8SC02239A.

[38] M. Karasuyama, H. Kasugai, T. Tamura, K. Shitara, Computational design of stable and highly ion-conductive materials using multi-objective bayesian optimization: Case studies on diffusion of oxygen and lithium, Computational Materials Science 184 (2020) 109927. doi:10.1016/j.commatsci.2020.109927.

[39] B. Letham, E. Bakshy, Bayesian Optimization for Policy Search via Online-Offline Experimentation, arXiv:1904.01049 [cs, stat] (2019). arXiv:1904.01049.

[40] J. Qu, A. Balvanz, S. Baranets, S. Bobev, P. Gorai, Computational design of thermoelectric alloys through optimization of transport and dopability, Mater. Horiz. (2022) 10.1039.D1MH01539G. doi:10.1039/D1MH01539G.

[41] R. Tanabe, H. Ishibuchi, An Easy-to-use Real-world Multi-objective Optimization Problem Suite, Applied Soft Computing 89 (2020) 106078. doi:10.1016/j.asoc.2020.106078. arXiv:2009.12867.

[42] B. Yan, Z. Wang, T. Fa, L. Cheng, P. Zhang, Bi-objective optimization of thermal conductivity and thermal stress of UO2–Mo–Nb composite through Gaussian process regression and NSGA-II method, Ceramics International (2022) S027288422201584X. doi:10.1016/j.ceramint.2022.05.031.

[43] L. M. Zintgraf, D. M. Roijers, S. Linders, C. M. Jonker, A. Nowé, Ordered Preference Elicitation Strategies for Supporting Multi-Objective Decision Making, arXiv:1802.07606 [cs, stat] (2018). arXiv:1802.07606.

[44] A. Zweig, G. Chechik, Group online adaptive learning, Mach Learn 106 (2017) 1747–1770. doi:10.1007/s10994-017-5661-5.

[45] S. G. Baird, M. Liu, T. D. Sparks, High-dimensional Bayesian optimization of 23 hyperparameters over 100 iterations for an attention-based network to predict materials property: A case study on CrabNet using Ax platform and SAASBO, Computational Materials Science 211 (2022) 111505. doi:10.1016/j.commatsci.2022.111505.