ChemOS 2.0: an orchestration architecture for chemical self-driving laboratories

Malcolm Sim[1,2], Mohammad Ghazi Vakili[1,2], Felix Strieth-Kalthoff[1,2], Han Hao[1,2], Riley J. Hickman[1,2,3], Santiago Miret[4], Sergio Pablo-García[1,2,*], and Alán Aspuru-Guzik[1,2,3,5,6,7,8,*]

[1]Department of Chemistry, University of Toronto, Lash Miller Chemical Laboratories 80 St. George Street, ON M5S 3H6, Toronto, Canada
[2]Department of Computer Science, University of Toronto, Sandford Fleming Building, 40 St. George Street, ON M5S 2E4, Toronto, Canada
[3]Vector Institute for Artificial Intelligence, 661 University Ave. Suite 710, ON M5G 1M1, Toronto, Canada
[4]Intel Labs, 2200 Mission College Blvd, Santa Clara, CA 95054, USA
[5]Department of Materials Science & Engineering, University of Toronto, 184 College St., M5S 3E4, Toronto, Canada
[6]Department of Chemical Engineering & Applied Chemistry, University of Toronto, 200 College St. ON M5S 3E5, Toronto, Canada
[7]Lebovic Fellow, Canadian Institute for Advanced Research (CIFAR), 661 University Ave., M5G 1M1, Toronto, Canada
[8]Acceleration Consortium, 80 St George St, M5S 3H6, Toronto, Canada
*Corresponding authors email: spgarcica@gmail.com, alan@aspuru.com

August 4, 2023

Abstract

Self-driving laboratories (SDLs), which combine automated experimental hardware with computational experiment planning, have emerged as powerful tools for accelerating materials discovery. The intrinsic complexity created by their multitude of components requires an effective orchestration platform to ensure the correct operation of diverse experimental setups. Existing orchestration frameworks, however, are either tailored to specific setups or have not been implemented for real-world synthesis. To address these issues, we introduce ChemOS 2.0, an orchestration architecture that efficiently coordinates communication, data exchange, and instruction management among modular laboratory components. By treating the laboratory as an "operating system" ChemOS 2.0 combines ab-initio calculations, experimental orchestration and statistical algorithms to guide closed-loop operations. To demonstrate its capabilities, we showcase ChemOS 2.0 in a case study focused on discovering organic laser molecules. The results confirm the ChemOS 2.0's prowess in accelerating materials research and demonstrate its potential as a valuable design for future SDL platforms.

1 Introduction

Global humanitarian and ecological challenges have sparked an unprecedented demand for novel functional materials across diverse industries, including clean energy technologies (renewable energy conversion and energy storage) [1] as well as medicine and health care.[2] Motivated by the urgency of these crises, researchers have realized the need to accelerate the often laborious and empirical discovery process of designing, fabricating and testing new materials.

Given this need, recent research efforts have shown remarkable progress in automated experimentation for various steps of the materials discovery cycle, including synthesis, formulation and device fabrication, functional characterization, and computational simulations. Notable advances in this regard range from early examples of automated biomolecule synthesis,[3] flow chemistry and microfluidics for solution-phase synthesis [4, 5], high-throughput experimentation for biological assays or reaction screening,[6, 7] to automated systems for fabricating solid-state or thin-film materials,[8, 9] as well as automated computational tools [10–12] and the application of big data [13] for virtual screening. The merger of such automation platforms with advances from "Artificial Intelligence" (AI) has given rise to the concept of "self-driving laboratories" (SDLs): the closed-loop integration of data-driven experiment planning with automated experiment execution. Such autonomous experimentation systems have been successfully demonstrated to address diverse optimization problems in a sample-efficient (and thereby time-efficient) manner, significantly reducing the required experimental resources.[14] While

early examples have mainly focused on reaction condition optimization, recent SDLs have shown the potential to discover new materials compositions, e.g. for quantum dot synthesis,[15] thin-film devices,[8] nanoparticles [16] or solid-state materials.[17]

Whereas automated systems have enabled increased throughput and enhanced reproducibility compared to human experimentation, their flexibility and reconfigurability has remained limited. Whilst this can be partly attributed to the often human-centric hardware design, software integration and the dynamic orchestration of automated workflows have remained a major challenge in automated laboratories. To address this challenge, a range of software solutions such as ChemOS,[18] AresOS,[19] NIMS-OS,[20] among others,[21–27] have been proposed for workflow management and hardware integration, with more ambitious approaches implementing complete frameworks combining hardware and software,[28–30] and standardizing experimental protocols.[31] However, existing frameworks overlook how researchers automate hardware, and do not provide flexible human-robotic integrations or a proper universal framework. Moreover, current frameworks do not incorporate IoT breakthroughs like fog computing, which offer strong data management and storage without relying on remote servers.[32, 33] Failing to embrace these cutting-edge technologies is a missed opportunity for optimizing SDL performance and making them more usable for experimental scientists.

Additionally, the lack of integration between experimental and computational simulations hampers the research capabilities of many laboratories by not incorporating theoretical insights provided by computations. An ideal

orchestration framework should tackle all of aforementioned challenges to promote widespread adoption and continuous improvements of diverse kinds of SDLs.

Remarkably, the current state of the automated laboratory landscape shows notable parallels to the early era of computer development in the 1950s and 1960s, where a multitude of specific devices, processing units, and workflows were developed with little flexibility and interoperability. Historically, this has inspired the development of standardized, portable operating systems (OS), which form the backbone of modern computer technology, and have enabled the widespread adoption of computer systems. The UNIX OS,[34] arguably the most prominent class of operating systems, are based on three simple design concepts: (i) the development of small functional units for specific tasks, (ii) the robust inter-operation of these units and (iii) their inter-communication through plain text streams.[35]

Based on this UNIX philosophy, we propose ChemOS 2.0 (**Figure 1**), a versatile, adaptable and portable "Operating System" for automated laboratories, fostering a holistic approach to operate and orchestrate various experimental and computational units. In analogy to computational operating systems, standardized plain-text communication protocols allow not only for the flexible integration of new software and hardware units, but enable the seamless interaction with human researchers. At the heart of our architecture is the integration of a fog computing device, offering improved efficiency and faster decision-making within the laboratory. Much like a traditional OS kernel, the fog computing device manages data, coordinates components, and empowers the lab to operate autonomously. We first describe the general

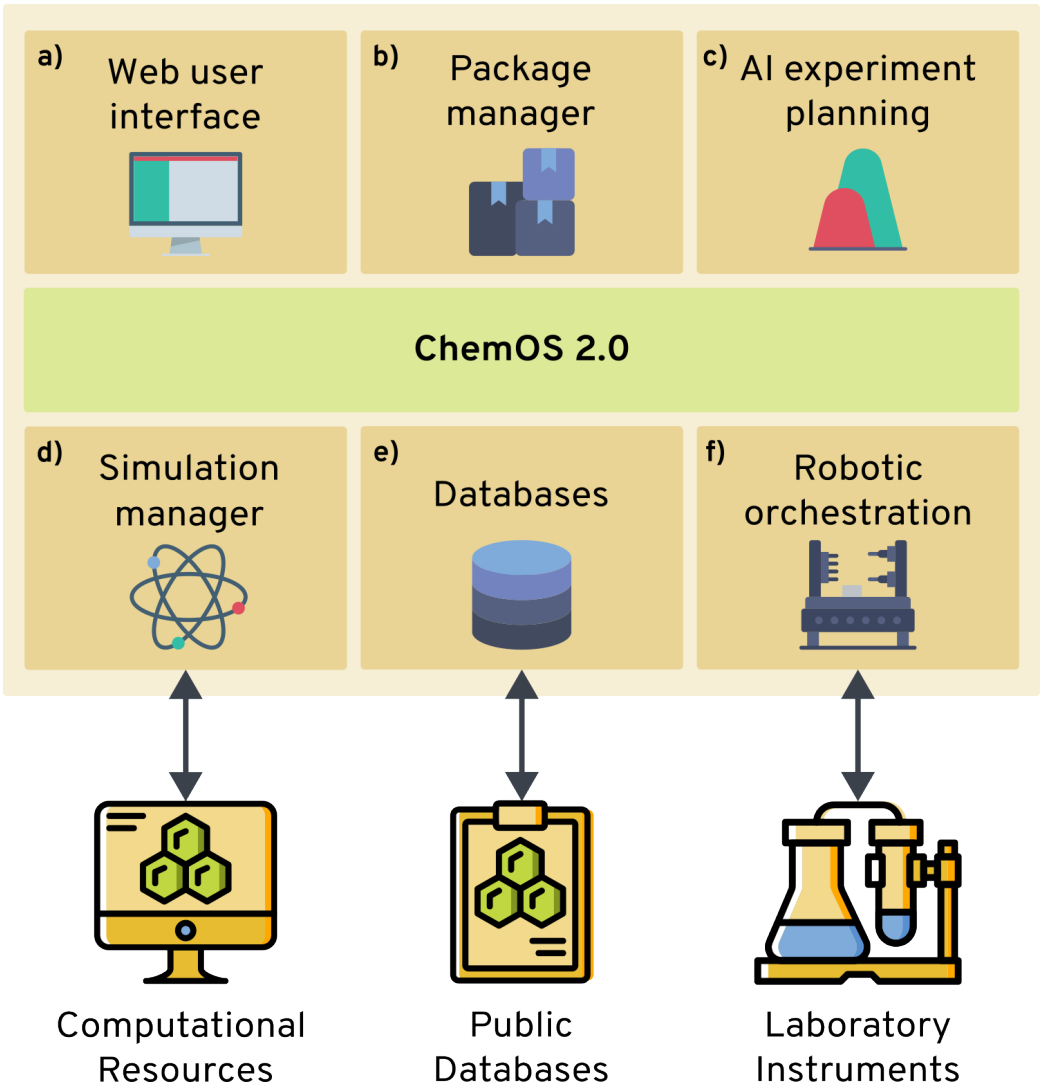

Figure 1: Features and capabilities of ChemOS 2.0: **a** web graphical interface to ease the user interaction, **b** package to assure full software reproducibility, **c** Bayesian optimizer platform for experimental planning, **d** DFT workflow manager connected to our high-performance computer cluster to orchestrate ab-initio DFT experiment, **e** SQL database server hosting both experimental and simulation databases and **f** Communication protocol to control laboratory instruments. **a,d-f** modules enable the communication layer with external agents.

architecture of ChemOS 2.0 along with its main design elements. Second, we demonstrate the applicability of ChemOS 2.0 in our own laboratory, showcasing the full orchestration of a complex materials discovery workflow towards novel gain materials for organic solid-state lasing devices.

2 Results

2.1 Software management

Meticulous control over the laboratory's software ecosystem is imperative to enhance experimental reproducibility, increase transparency, and mitigate production failures. As such, achieving complete transparency necessitates stringent management of the software state within a laboratory with dependency conflicts due to incompatible software versions posing a well-recognized challenge in this regard.[36] To address these concerns, the core design of ChemOS 2.0 incorporates an orchestration fog device that runs the necessary software layers for laboratory operations while keeping each of the laboratory modular for streamlined integration. To ensure reproducibility, robustness, and seamless deployment, the fog orchestration platform is equipped with NixOS, a declarative package manager-based (Nix) operating system known for its ability to provide precise control over system state and software versions.[37] Nix provides precise control over the system state by enabling an accurate selection of software versions and dependencies. Defining local and global system states in NixOS revolves around the use of `.nix` files, containing a declarative system state recipe. Researchers can use configuration files

to share digital experimental environments and deploy pre-configured operating systems on different platforms. Thus, `.nix` files are pluggable modules, whose addition or removal will change the state of the system. While some of the code may be hardware-specific, these files serve as a foundation for building unique SDL architectures. In our work, we have prepared a collection of `.nix` files containing the environmental state used for our experiments. The configuration code and installation instructions can be found in the ChemOS 2.0 repository.[38]

2.2 Data management

Ameliorating experimental reproducibility requires a proper data management plan involving the collection of experimental procedures, outcomes, and environment metadata from an SDL. Notably, we draw inspiration from the remarkable advancements in computational chemistry, where sophisticated data solutions have successfully ensured reproducibility in chemical simulations.[13, 39–43]

To streamline data management, our orchestration device is equipped with two independent relational databases: one for experimental, and another for simulation data. The first is designed to store information about the laboratory state, including raw and processed data and metadata of the instruments, as well as the input and output of experimental processes. We implement this desired modularity in the database layout by defining key universal sub-tables, `device`, `job`, and `devicelog`, and more specialized ones tailored to the needs of each instrument. This concise layout ensures max-

imum flexibility and a robust architecture for experimental data collection, that is readily transferable to any SDL. A detailed layout of the experimental database can be found in **Note S1** and **Figure S3**. On the other hand, the simulation database is managed by AiiDA, which provides a well-tested and robust design specifically tailored for simulation orchestration.[12] These databases are not only used to keep track of all the data collected in the laboratory, but they also serve as a broker to share data between different devices and software running in the laboratory. ChemOS 2.0's databases can be easily exported and published into specialized databases to report complete experimental information. As such, we gain a more comprehensive snapshot of the laboratory, encompassing all instructions executed by experimental devices and their corresponding metadata.

2.3 Device communication

Enabling seamless communication between hardware and applications is a crucial role of an operating system, particularly in the context of self-driving laboratories, where applications can be likened to chemical procedures with multiple steps. In ChemOS 2.0 the interaction between the system and hardware is facilitated through a core middleware software unit, operating on the terminals physically connected to the devices and the local network, as depicted in **Figure 2** with the SiLA2 client/server standard managing the communications.[27] In our implementation, the instruments themselves are treated as SiLA2 servers, exposing all their valid actions to the local network, while our orchestration device acts as a client capable of querying the

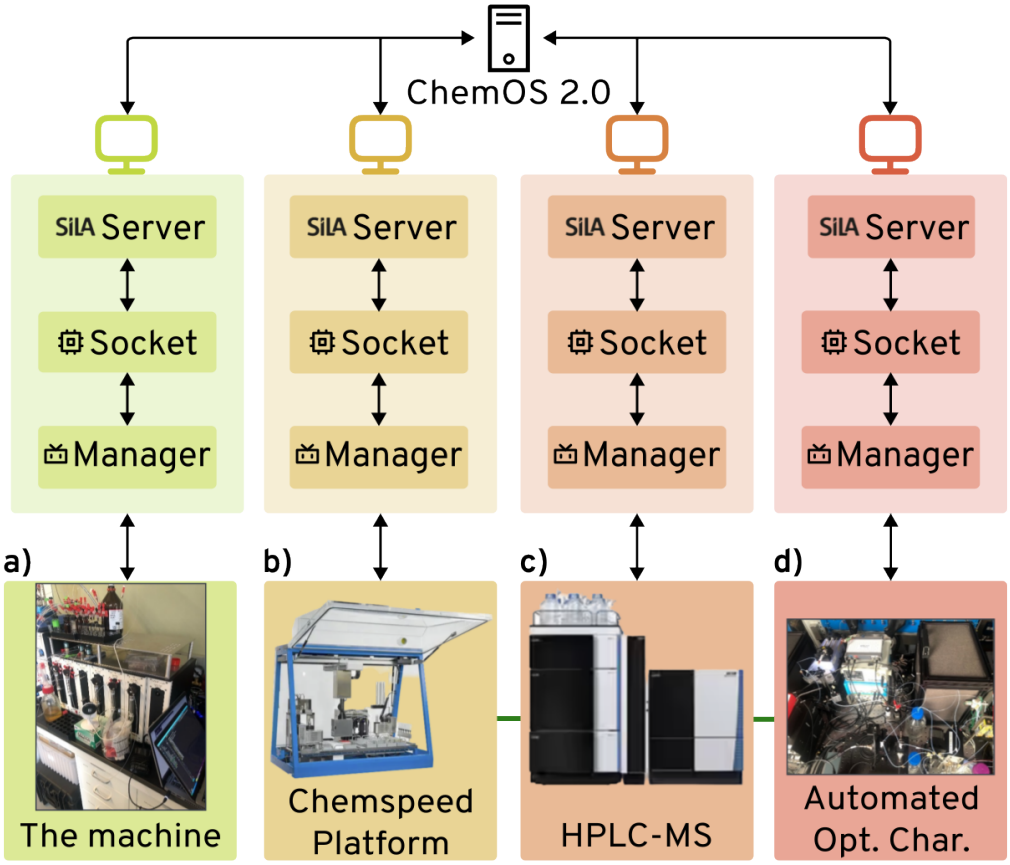

Figure 2: The self-driving lab is orchestrated using ChemOS 2.0, which leverages the SiLA2 client/server protocol to control laboratory devices. The client interacts with the servers on the network to access available actions. Each server operates as a daemon on the respective computer connected to the laboratory instrument. In our configuration, device-specific managers also run as daemons, providing a GUI for user interaction with the instrument. Communication between the SiLA2 servers and managers is facilitated through a computational socket, enabling command transmission via ChemOS 2.0 or the GUI. This setup empowers users to command various laboratory instruments: **a** *The Machine*, robotic organic synthesis platform [44], **b** Chemspeed platform, a robotic general synthesis platform, **c** an HPLC-MS module for characterization and **d** our custom-built automated optical characterization able to characterize lasing properties. Instruments **b-d** are physically connected, thus enabling direct transfer of materials.

devices to execute their operations. The SiLA2 package comes with a suite of commands that generate, deploy and maintain the available actions of the SiLA2 servers by defining `XML` files. Details of using SiLA2 in python can be found in **Note S2**, and an example `XML` file can be found in **Note S7**. In adherence with the third UNIX principle, job files to the SiLA2 servers are all in a human-readable `json` format for all of the instruments (See **Note S15**). The manager component often serves as a device driver, and often includes highly abstracted code capable of executing complex actions. Commercial solutions usually consist of graphical user interfaces (GUIs) tailored for individual operations and often lack an application programming interface (API) for direct communication.[45] Consequently, custom software modifications are often required to exert control over these devices. The manager therefore provides a "workaround" for some instruments that are limited by their provided user interfaces for automation, or a device driver for more barebones/simplistic instruments. To bridge the communication gap between the (standardized) SiLA2 server and the (device-specific) manager, we implemented a socket server. Although the decoupling introduced by this connection adds complexity to the code architecture, the socket communication layer acts as a safeguard in case of a loss of connection during the execution of sensitive/hazardous operations. The socket layer also addresses the practical need for our researchers to sometimes operate instruments without ChemOS 2.0, and critically, ensure the continued operation of the instrument in the event of a loss of connection between ChemOS 2.0 and the instrument. A deeper description of the SiLA2 server implementation and the socket server can be found in **Notes S8, S9, S10** and **Figures S1, S2,**

S5, respectively. A code example of interacting with a SiLA2 server can be found in **Note S11**.

Lastly, we have successfully deployed the aforementioned middleware unit in our experimental laboratory. **Figure 2** illustrates our current setup, which includes: (a) *The Machine* (**Note S6**): a robotic pump system capable of parallelized synthesis in solution,[44] (b) Chemspeed platform, an automated robotic framework for parallelized synthesis and characterization workflows (**Figure S3**),[46] (c) a High-Performance Liquid Chromatography/Mass Spectrometetry (HPLC-MS) device (**Note S4**) for compound separation and identification, and (d) the custom-built automated optical characterization: a flow setup for automated spectroscopy [47] (**Note S5**). The available commands for each of the experimental units can be found in **Tables S1-S5**. To test the SiLA2 server before deployment, we used "simulation" versions of the workflow managers (**Note S8**), ensuring minimal interruptions of operations while the server is in development. The operations of each instrument are controlled by an independent computer connected to the local network. The instruments of the laboratory can be used in-sequence or individually.

2.4 DFT integration

While experimental laboratories contribute significantly to scientific research, the forefront of chemical exploration relies on ab-initio simulations to attain profound insights into complex chemical processes. Many of these simulations, however, come with substantial computational demands and require

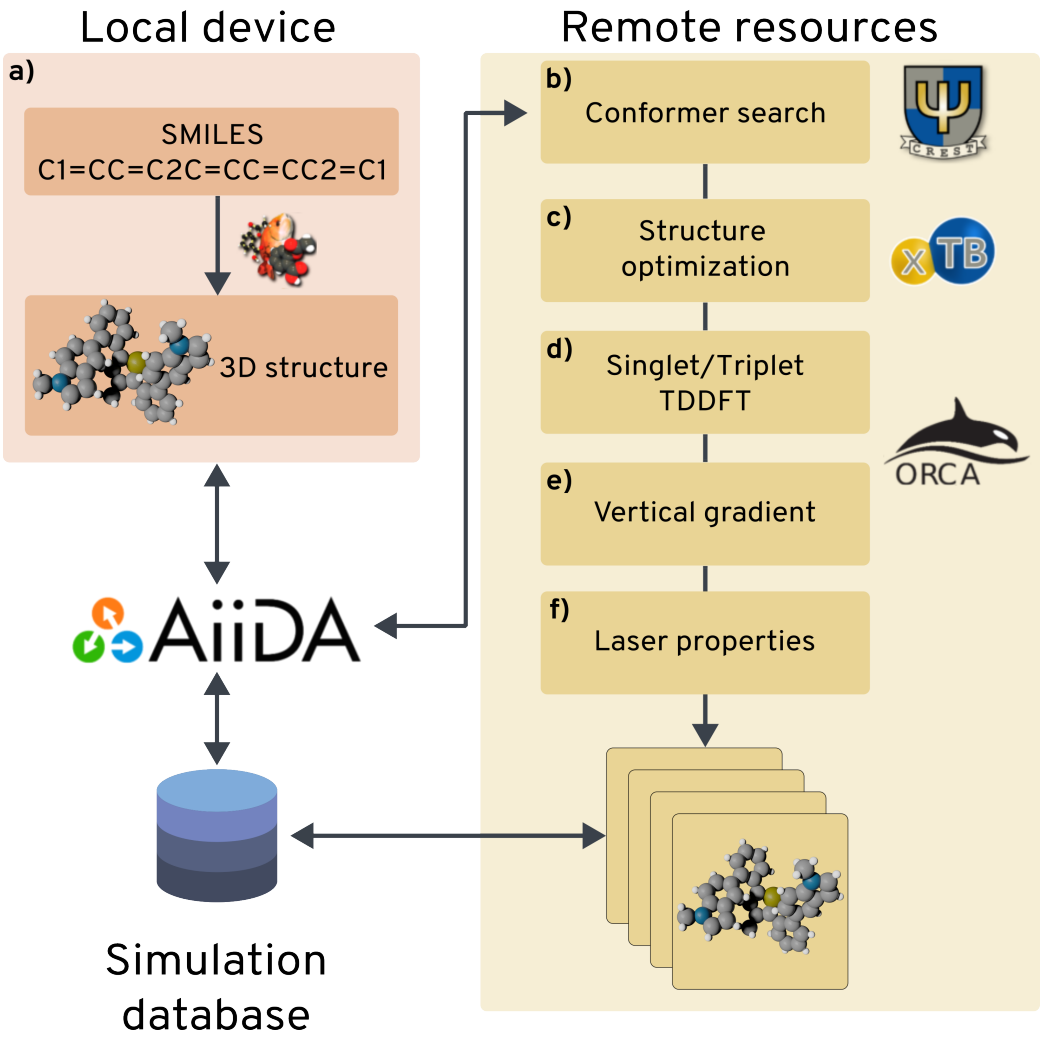

Figure 3: **a** building a 3D structure of the molecule from a SMILES string using Open Babel,[48] **b** searching for the lowest-energy conformer using the crest package,[49] **c** optimizing the ground state structure and calculating the corresponding Hessian of the lowest-energy conformer using the xTB-GFN2 semi-empirical Hamiltonian,[50] **d** evaluating the ground and excited state energies and gradient of both singlet and triplet states at the time-dependent DFT level using ORCA,[51] **e** determining the vibrationally coupled absorption and emission spectra using the vertical gradient approximation and **f** computing a proxy for lasing properties.[52]

specialized external computational facilities for execution. The need for simulations in chemical exploration to improve the quality and efficiency of research makes it crucial to equip ChemOS 2.0 with a dedicated unit capable of handling such chemical simulations.

To tackle the computational challenges and ensure reproducibility, ChemOS 2.0 embraces the integration of the AiiDA software package[12]. AiiDA plays a critical role in automating data transfers between the user's local environment and the high-performance supercomputing cluster. By utilizing the SSH protocol, AiiDA submits calculations to a SLURM queue, retrieves outputs, and stores the results in a centralized database. It also resolves relevant dependencies between the different steps, helping to properly schedule their execution. To ensure reproducibility, AiiDA stores the data and metadata for each step of the workflow in the simulation database and provides Python bindings to ease the query. Additionally, AiiDA is prepared for distributed computing, enabling efficient utilization of computational resources. As a result, lightweight operations such as 3D molecular structure generation can be performed locally, while computationally intensive tasks utilizing packages like ORCA [51] and xtb [50] can be executed on our high-performance supercomputing cluster.

In this work, we leverage the capabilities of AiiDA to create a powerful DFT workflow for estimating the lasing performance of organic laser molecules. Taking a SMILES string as input, the AiiDA-enabled workchain efficiently executes the necessary calculations, monitors their status, and retrieves critical data related to the lasing performance. This data is subsequently used in a Bayesian optimization campaign, enabling researchers to

make informed decisions and advance their chemical exploration efforts. The workflow is depicted in **Figure 3** and additional information can be found in the **Methods section**.

2.5 Web interface

ChemOS 2.0 operates as a server-based system, primarily relying on usage via command-line interactions. While this approach is well-suited for specialists, it may not be the preferred method for chemists who are more accustomed to graphical user interfaces (GUIs).

To facilitate the adoption of our system, we developed a user-friendly web application interface, empowering any researcher to send experimental jobs easily, monitor their progress, and analyze the resulting data. This layout is shown in **Figure S4**. The web application is built using the Streamlit package [53], is hosted on the orchestration device and includes a sidebar menu that enables toggling between the available devices of the laboratory. Once the user clicks into a tab, the web interface shows the actions of the device and the inputs needed for their execution. Users can then upload job files, visualize and download results, and control the Bayesian optimizer using the application. Workflows involving multiple steps need to be coded in Python using scripts that send commands to the instruments and modules. Moreover, the SiLA2-type system offers a promising approach for implementing a no-code platform, enabling users to define workflows in a more user-friendly manner.

Overall, the introduction of this web application interface significantly

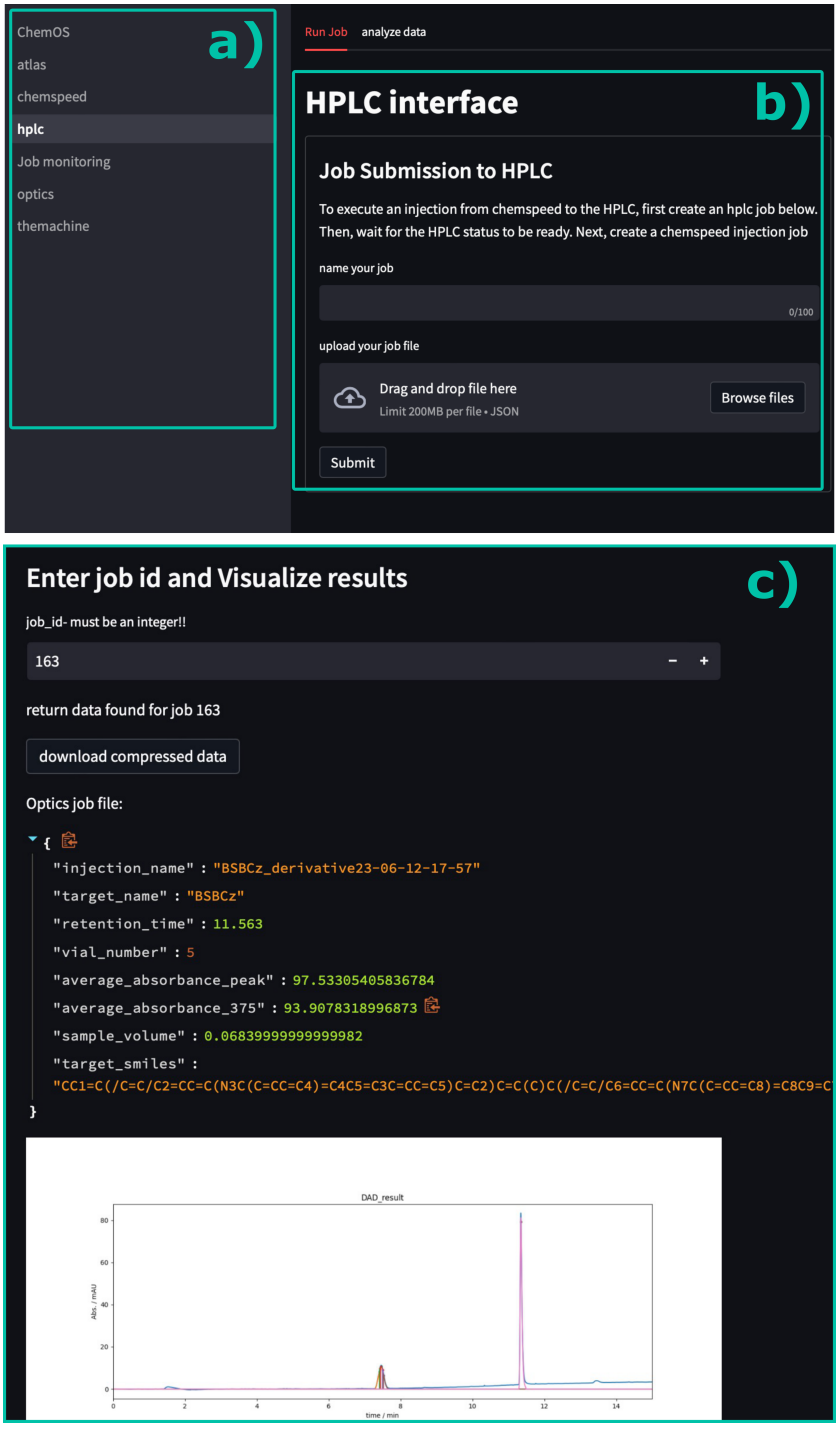

Figure 4: Web application for ChemOS 2.0. **a** sidebar for toggling between instruments/features. **b** job submission for the HPLC-MS. **c** data visualization of the HLPC-MS's job results.

simplifies the usability of our system and enhances its accessibility for researchers across diverse skill levels. In this work, we have implemented functionality for all the available hardware presented in **Subsection 2.3**. While a full-featured web interface would enhance the capability of ChemOS 2.0, its development would require substantial efforts from additional fields and is beyond the scope of this study.

2.6 Experimental planner

Experimental planners are the keystone of SDL prompting us to equip ChemOS 2.0 with Atlas,[54] a package for Bayesian optimization (BO) designed explicitly for experimental sciences. Atlas is a general-purpose optimization framework for expensive-to-evaluate black-box problems, capable of mixed-parameter and multi-objective optimization. Importantly, Atlas incorporates BO concepts particularly relevant to chemistry, such as a molecular kernel function,[55] general experimental parameter optimization,[56] robust optimization,[57] and asynchronous experimental execution.[58, 59] For more information, see **Note S12** and the Ref.[54], where we show an additional closed-loop electrochemical campaign orchestrated by ChemOS 2.0.

2.7 Experimental campaign

To demonstrate a proof-of-concept of a fully closed loop of automated experiment planning and execution, we designed a multi-objective campaign for the synthesis of Bis[(N-carbazole)styryl]biphenyl (BSBCz) derivatives based on previous work.[47] The campaign goals are to simultaneously maximize

the experimental gain cross section and the simulated lasing gain factor (produced via DFT simulations), involving BO-based experiment planning, automated synthesis, purification and functional characterization, as well as simultaneous computational simulation via the AiiDA module.

The search space of potential target molecules in this campaign is defined by enumerating all products of a double Suzuki–Miyaura coupling between the MIDA ester (MIDA = N-methlyiminodiacetic acid) of (E)-2-(4-(N-Carbazoly)phenyl)-vinyl-1-boronic acid and 38 commercially available dihalides.[47] A detailed synthesis scheme can be found in **Figure S13**. This space is navigated by Atlas using a gaussian process surrogate model [60] with a Tanimoto similarity kernel function.[55] Initial recommendations were generated by training the optimizer on two data points from the original work. The recommended target molecules are then synthesized and characterized experimentally, while spectroscopic properties are simulated computationally through the aforementioned workflow. We provide Atlas with the goal of maximizing both the simulated spectral gain factor (from the AiiDA workflow), as well as the experimental gain cross section via an equally weighted expected hypervolume improvement acquisition function that jointly maximizes all objectives.

2.8 Testing hardware functionality with *The Machine*

Before creating a fully closed-loop, we evaluated the experimental capabilities of our setup by performing human-in-the-loop orchestration test excluding an experimental planner or DFT simulations. The test involved the synthe-

sis, characterization, and measurement of optical properties for a compound recommended by our experimental planner.

The synthesis was performed using *The Machine*. During the test, ChemOS 2.0 directed the synthesis of the suggested BSBCz derivative as recommended during the creation of the BO campaign by Atlas. The synthesis instructions were seamlessly communicated to *The Machine* through its SiLA2 server, and real-time data were logged in the experimental database.

Following synthesis, the crude reaction mixture was manually filtered and transferred to the HPLC-MS device for characterization. Purification (HPLC-MS) and spectroscopic characterization (custom-built automated optical characterization platform) were controlled by ChemOS 2.0 through SiLA2 commands. The desired product was successfully detected by HPLC-MS, and the purified product underwent downstream optical property analysis. All data produced during this test can be found in **Note S14**.

2.9 Building a closed-loop workflow with the Chemspeed platform

Lastly, we deployed and executed a closed-loop workflow for the synthesis of BSBCz derivatives. In addition to orchestrating the experimental component of the campaign, we conducted DFT simulations parallel to satisfy the two objectives required by the BO campaign **Subsection 2.6**. Details can be found in **Note S16** and **Figure S16** For the closed loop case study, we used Chemspeed to synthesize the BSBCz derivative. Creating the closed loop encompassed sending the SiLA2 commands to the SiLA2 servers.

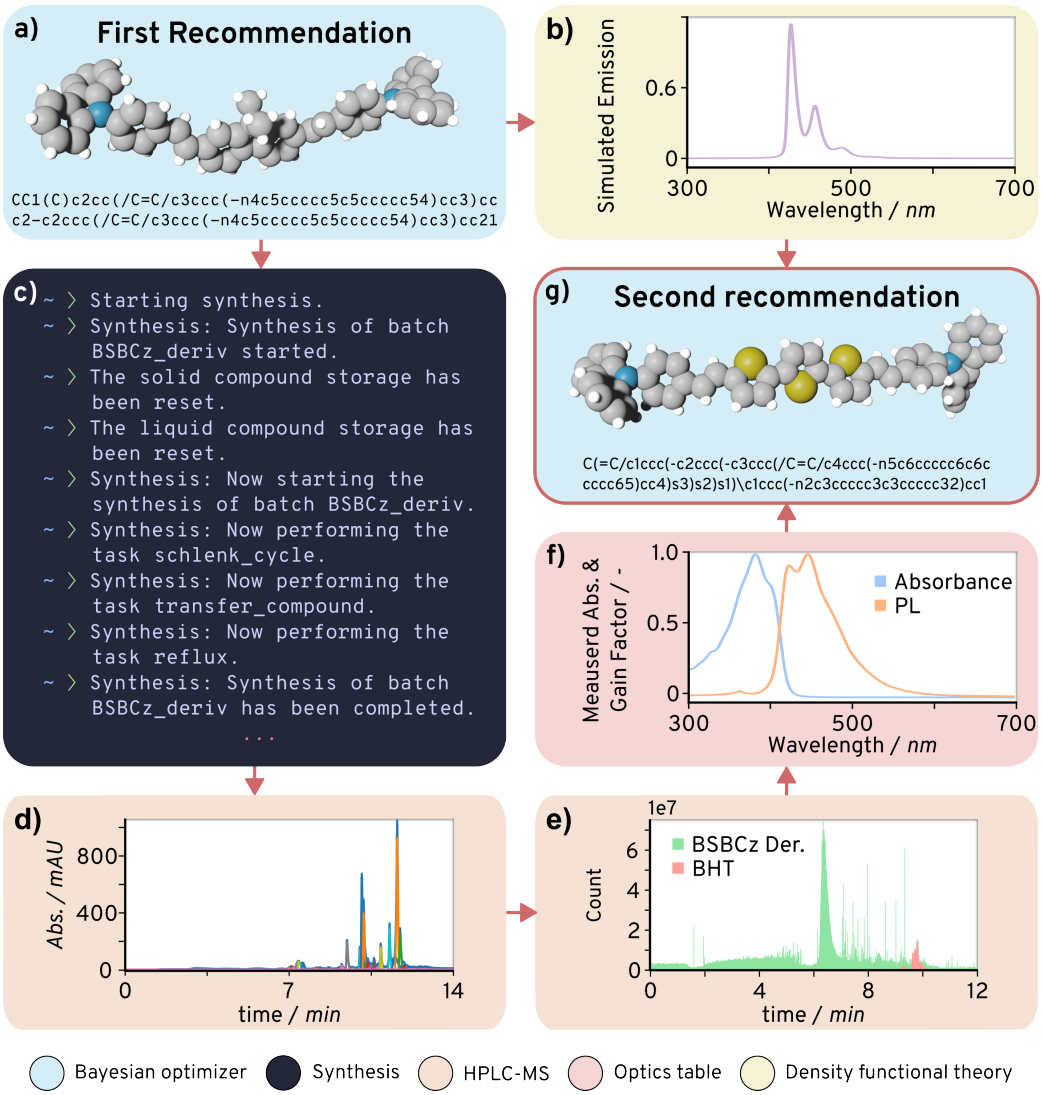

Figure 5: Results saved on ChemOS 2.0 for the synthesis of BSBCz derivatives. **a** First recommendation candidate made by the experimental planner. **b** Simulated fluorescence spectrum of BSBCz derivative produced by DFT workflows on ChemOS 2.0. **c** Logging information for organic synthesis on ChemSpeed. **d** Diode Array detector (DAD) chromatogram and **e** Extracted Ion Chromatogram (XIC) plots for the HPLCMS. **f** Time-Correlated Single Photon Counting and absorbance/fluorescence spectra of the custom-built automated optical characterization platform analysis. **a** Second recommendation made by the experimental planner based on the results of **b** and **f**. All results and raw data files are stored in the database during the execution of experiments.

The results and workflow are illustrated in **Figure 5** showing the following steps: (a) ChemOS 2.0 queries the Atlas Bayesian optimizer with two literature-known "seed" observations and receives a recommendation for which BSBCz derivative to synthesize. (b) Then, it instructs Chemspeed to execute the synthesis. (c) When the reaction is complete, ChemOS 2.0 sends the compounds to the HPLC-MS to detect the product in the crude mixture and transfers the purified BSBCz derivative to the custom-built automated optical characterization platform. (d) Finally, the optical characterization platform measures sample fluorescence and absorbance spectra and the emission lifetime to compute the gain cross section. (e) In parallel to the experimental loop, the computational job is sent to the AiiDA daemon to obtain the simulated spectral gain factor. (f) The results from the two previous steps are used to re-train the Bayesian optimizer and query another candidate. All logging information was stored in the databases using the database architecture described in **Subsection 2.2**. The experimental gain cross section value was estimated experimentally using fluorescence spectra, relative quantum yields, and emission rates of the BSBCz derivatives using our automated optical characterization platform. Details of the calculation can be found in previous work.[47]

In summary, ChemOS 2.0 was able to successfully orchestrate a full SDL loop, including manipulation of different instruments, an AI experiment planner, and stored all output data in an internal database.

3 Discussion

We demonstrate that more general, transferable and multifaceted SDLs are possible thanks to central lab orchestration. The modular nature of ChemOS 2.0 enables laboratories from different fields to configure it according to their domain specific requirements.

Following the UNIX philosophy, ChemOS 2.0 acts an "operative system", with tools and instruments working together to create workflows and present data in a human-readable format. Unlike monolithic designs in other orchestration configurations, our approach reflects an operating system's design philosophy, promoting flexibility and allowing users to easily modify layers and incorporate new features into the setup, such as new instruments and experiment planners. The database design follows the same principles, allowing users to rapidly add new devices to automated workflows and data collection schemes using a universal-specialized table scheme, as well as device-specific additions. Thus, rather than building a framework specific to our setup, we have designed an architecture amenable to this experimental lab along with a diverse set of laboratory settings. Moreover, by employing a central fog orchestration device as the core of our architecture, we ensure greater robustness and adaptability, allowing an automated management of laboratory services and ensuring software reproducibility.

One of the remarkable advantages of our modular architecture is its support for flexible human/robot integration of automated tools. Instead of demanding a complete overhaul of laboratory layout and hardware, ChemOS 2.0 allows the automation of existing tools and minimally interrupts existing

workflows. A drastic transformation of existing labs is often impractical, as it disrupts the research activity of academics and industry professionals. This can discourage them from transitioning to digitization. ChemOS 2.0 has been designed with a hybrid integration in mind, allowing researchers to gradually introduce automation into their workflows. Initially, the Nix configuration can be installed on a workstation in the laboratory and be later expanded upon. Users can begin by integrating simpler devices, before adding instruments one by one to their ChemOS 2.0 ecosystem. While testing connectivity between ChemOS 2.0 and the instrument servers, simulators were used instead of the actual lab hardware. As such, the deployment of the SiLA2 servers and testing of the actual hardware was therefore minimized, and regular operations of the laboratory were minimally interrupted.

Digitization, including the implementation of tools like ChemOS 2.0, represent a considerable challenge for many laboratories, as it requires significant experience in the efficient use and integration of computational resources. We believe this creates the need for a new, emerging role in research laboratories: a "digital manager", responsible for orchestration, workflow assembly, and the integration of new (digital) tools, which we believe will be crucial to the proliferation of self-driving laboratories. Moreover, while is a financial barrier for automating many laboratories, we remain optimistic for the future given that the cost of automated tools is steadily decreasing each year.[61] Following digital managers and cost, another hurdle may be less straightforward to address: the API-type access to automated tools. To aid the proliferation of SDLs, manufacturers should comprehensive support for the automation of their hardware, and recognize that in doing so, they can actually encourage

more adoption of their products by academia, startups and industry.

Importantly, our work represents a prototype implementation of the discussed design principles, and leaves room for structural refinements and further software development. As self-driving laboratories become more prevalent, the connection between orchestration devices will become a significant consideration, presenting challenges related to scalability and cybersecurity. Furthermore, the increasing complexity of laboratory environments will necessitate the development of more powerful and user-friendly interfaces. Graphical interfaces and natural language processing,[62] enable users to communicate with the laboratory and coupled with advanced data analysis methods, will allow programming-averse users to leverage the features of the ChemOS 2.0 platform effectively.

4 Conclusion

In conclusion, this work represents an advancement in promoting the adoption of innovative digital tools by chemists in their laboratories. The integration of Bayesian optimization, experimental techniques, ab-initio simulations, and data management within ChemOS 2.0 unifies automated laboratories to conduct sophisticated workflows. Our hybrid approach not only relies on experimental results but also incorporates in-depth sample analysis through DFT calculations, enabling comprehensive data-driven research. The automation of data collection opens up exciting opportunities for machine learning and data science applications in chemistry, fueling accelerated research progress and discoveries. Looking ahead, we envision even greater

possibilities through enhanced communication and collaboration among multiple ChemOS 2.0 platforms, fostering seamless cooperation between different laboratories. ChemOS 2.0 lays a solid foundation for designing novel SDLs, making them more modular, robust and making their units transferable. This advancement supposes an step forward in the quest of revolutionize materials research through streamlined and efficient self-driving laboratories.

5 Methods

5.1 Experimental procedures

The synthesis of BSBCz derivatives was performed using the Chemspeed SWING XL automated platform, available at the University of Toronto. For more details of the automation manager for the Chemspeed platform, see **Note S3**. Automated Suzuki–Miyaura couplings were performed following the general slow-release conditions reported by Grzybowski, Burke and coworkers.[44] Synthesis instructions were received from the SiLA2 client on ChemOS 2.0. Reactions were performed in the ISYNTH reactor under a nitrogen atmosphere, established by five evacuate–backfill cycles. Reactants and reagents were transferred as stock solutions in 1,4-Dioxane or water, respectively under a constant stream of nitrogen, using the 4-Needle liquid dispensing tool. The reaction mixture was heated to 100 °C under vortex stirring for 12 h. The reaction mixture was cooled down to room temperature for 1 h, and the crude reaction mixture was filtered over a short plug of celite (using the Solid Phase Extraction module of the Chemspeed SWING

XL platform), eluting with tetrahydrofuran and collecting the eluate. Following communication with the HPLC-MS device through the respective SiLA2 client on ChemOS (details see below), the eluate is automatically injected to the HPLC-MS using the on-deck injection valve equipped with a sample loop.

5.1.1 Separation and identification in HPLC-MS

Analysis of crude reaction mixtures and product isolation was performed by HPLC-MS on a ThermoFisher system, equipped with a Vanquish HPLC and a QExactive mass spectrometer. Injection to the HPLC-MS system was realized through a 6-port injection valve on the deck of the Chemspeed SWING XL platform. Job submission and orchestration of operations between the two devices was enabled by ChemOS 2.0. Upon submission of a job from ChemOS 2.0 to the HPLC-MS device, an HPLC-MS run is automatically started. After the equilibration stage, ChemOS 2.0 can read the *ready* status from the HPLC-MS device, and commence the analyte transfer into the sample loop on the Chemspeed SWING XL. Data acquisition on the HPLC-MS device is started through an electronic trigger sent from the injection valve driver on the Chemspeed SWING XL, indicating the exact sample injection time. Upon completion of the first analysis run, the target compounds are identified by matching peaks in the wavelength-averaged UV/vis chromatogram and their extracted ion chromatograms, respectively. The detailed workflow for peak identification is described in previous work.[47] All analysis results are stored on ChemOS 2.0. In case of successful detection of the desired reaction product, a second HPLC-MS run is initialized by ChemOS 2.0,

following the workflow detailed above. In this case, the fraction containing the desired product is automatically collected using a downstream selection valve, and is used for spectroscopic characterization (see below).

5.1.2 Custom-built automated optical characterization

The collected sample is automatically characterized in the spectroscopic workflow described in **Note S5**, orchestrated by ChemOS 2.0. Upon successful completion of the HPLC-MS run and automated fraction collection, spectroscopic analysis can be initiated from ChemOS 2.0. On the custom-built device, the sample is transferred to different flow cuvettes using a syringe pump. Steady-state absorption spectra are recorded using a white-light lamp (Ocian Insight DH-MINI) combined with a spectrometer (Ocean Insight, QEPro XR). Steady-state emission spectra were recorded on the same spectrometer using a 365-nm solid-state LED (Thorlabs, M365FP1 with DC4100HUB) as the excitation source. Quantum yields were determined by integration of the emission spectrum, while simultaneously recording the excitation transmission using a photodetector (Thorlabs, S120VC with PM100D). Quantum yields were referenced to a pre-measured sample of 4,4`-Bis[(N-carbazole)styryl]biphenyl ($\phi = 0.89$ in toluene).[63] Time-resolved emission spectra were recorded using time-correlated single-photon counting (TCSPC, PicoQuant TimeHarp 260 board) after excitation with a picosecond-pulsed laser (PicoQuant, LDH-D-C-375 with PL800-D), and the corresponding lifetimes were obtained by exponential fitting. Details on the data analysis workflow are described in Ref.[47]. Upon completion of the analysis workflow, all analysis results are automatically stored on

ChemOS 2.0.

5.1.3 The Machine

The Machine consists of syringe pumps and selection valves connected by tubing and further to solution and reaction vials and N_2 supply. The atmosphere handling was performed by inserting drawing gas from heal-space of vials solution consistently refilling N_2 and purge syringe connect to exhaust. The automated ligand transfer was performed by washing the syringe with desired solution by pumping from the source and dispensing to waste, followed by drawing a certain amount and dispensing to the destination vial. Finally, flushing were required drawing N_2 and flushing tubing content to destination. Each hardware execution was recorded and streamed as standard output. A final exit message was sent at the completion of the experiment.

5.2 Computational procedures

All SiLA2 servers were constructed using the Python implementation of the SiLA2 0.10.1 communication standard. The SiLA2 servers operate in a separate thread on the computers that control the driver for the instruments in the lab. Communication between the SiLA2 server and the instrument on the local computer is accomplished using either socket connections or via status folders and files. Meanwhile, all communication between the local instrument and other devices on the network is only done via the SiLA2 server.

ChemOS 2.0 uses NixOS 22.11 configuration files. Using this configuration file will produce an identical shell on any Linux device that has NixOS in-

stalled. The experimental database was constructed using SQLalchemy 1.4.41,[64] and the simulation database was built by AiiDA.[12] Both databases use PostGreSQL 14.6 as the backend. Interaction with the database on ChemOS 2.0 is accomplished through the use of psycopg2 2.9.5.[65] All storage of data in the database is done automatically through Python scripts executed on ChemOS 2.0. Connecting ChemOS 2.0 to the other devices (SiLA servers) in the lab is easy and straightforward. One has to create a client Python object and initialize it using the IP address and port of the targeted SiLA2 server. Afterwards, one can send any of the available commands to the SiLA2 server (the instrument), which are represented as methods of the `SilaClient` object. Calling a method of the client object returns a SiLA2 command instance. If a command is observable, it is possible to subscribe to the command and receive a stream of information as the command is executing. It is through this mechanism that output data from the instruments is streamed to the Fog server.

Tracking the progress of experimental campaigns with Atlas 0.0.0 is accomplished by using Olympus 0.0.1b0,[66] a benchmarking framework for Bayesian Optimization. Using Olympus' `Campaign` object, we can keep track of the search space and previous observations over the course of the optimization. To use the Atlas SiLA2 server The Olympus `Campaign` object is passed to the optimizer and ultimately to Atlas, which uses the `observations` attribute of the `Campaign` as input for suggesting new parameters. Saving the campaign state as a `.pkl` file is one way to track each successive step of the experiments. The Atlas SiLA2 server returns a parameter vector, which can then be added to the `Campaign`, and used to automate the next round of

experiments.

5.2.1 Density functional theory

Generation of molecular conformers was performed using crest 2.11.1.[49] Semi-empirical geometry optimization and frequency calculations of S0 and T1 structures were performed at gfn2-xtb level of theory using xtb 6.4.1.[50] DFT geometry optimization and frequency calculations of S0 structure were performed with ORCA 5.0.4 package,[51] at PBE0-D3/Def2-SV(P) level of theory. TD-DFT single-point and gradient calculations were performed with the ORCA 5.0.4 package, at PBE0-D3/Def2-SV(P) level of theory, with Tamm-Dancoff approximation. The total electronic energies of S0, S1-S5 and T1-T5, and the gradients of S0, S1-S3 and T1-T3 were computed at the DFT-optimized S0 structure. AiiDA 2.4.0 and AiiDA-shell 0.5.3 were used for workflow orchestration.[12, 67]

Data availability

As a showcase, the results of the DFT calculations have been uploaded to ioChem-BD.[39, 68]

Code availability

All the code needed for this work can be found in our Gitlab repository.[38] Tutorials can also be found to replicate the more generalizable parts of this work.

Acknowledgements

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Subaward by University of Minnesota, Project title: Development of Machine Learning and Molecular Simulation Approaches to Accelerate the Discovery of Porous Materials for Energy-Relevant Applications under Award Number DE-SC0023454. A.A.-G. acknowledges support from the Canada 150 Research Chairs Program as well as Anders G. Fröseth. Computations were performed on the Niagara supercomputer at the SciNet HPC Consortium. SciNet is funded by: the Canada Foundation for Innovation; the Government of Malcolm Sim acknowledges tremendous support from the Ontario Student Assistance Program's Ontario Graduate Scholarship; Ontario; Ontario Research Fund - Research Excellence; and the University of Toronto.

Author contributions

M. S., software, methodology, investigation, writing original draft. M. G. V., methodology, writing, conceptualization, architecture design. F. S.-K., preparation of experimental workflows, writing review & editing. H. H., preparation of simulation and experimental workflows. R. H., implementation of the experiment planner, writing review & editing. S. M., conceptualization, writing review & editing. S. P.-G., methodology, writing, conceptualization, supervision. A. A.-G., conceptualization, funding acquisition, supervision, project. administration, writing review & editing.

Competing Interests Statement

The authors declare the following competing interests

References

(1) Helm, D. (2012). The Kyoto approach has failed. *Nature 491*, Number: 7426 Publisher: Nature Publishing Group, 663–665, DOI: 10.1038/491663a.

(2) Wouters, O. J., McKee, M., and Luyten, J. Estimated Research and Development Investment Needed to Bring a New Medicine to Market, 2009-2018 | Drug Development | JAMA | JAMA Network.

(3) Merrifield, R. B., and Stewart, J. M. (1965). Automated Peptide Synthesis. *Nature 207*, 522–523, DOI: 10.1038/207522a0.

(4) Bédard, A.-C., Adamo, A., Aroh, K. C., Russell, M. G., Bedermann, A. A., Torosian, J., Yue, B., Jensen, K. F., and Jamison, T. F. (2018). Reconfigurable system for automated optimization of diverse chemical reactions. *Science 361*, 1220–1225, DOI: 10.1126/science.aat0650.

(5) Hartrampf, N., Saebi, A., Poskus, M., Gates, Z. P., Callahan, A. J., Cowfer, A. E., Hanna, S., Antilla, S., Schissel, C. K., Quartararo, A. J., Ye, X., Mijalis, A. J., Simon, M. D., Loas, A., Liu, S., Jessen, C., Nielsen, T. E., and Pentelute, B. L. (2020). Synthesis of proteins by automated flow chemistry. *Science 368*, Publisher: American Association for the Advancement of Science, 980–987, DOI: 10.1126/science.abb2491.

(6) Macarron, R., Banks, M. N., Bojanic, D., Burns, D. J., Cirovic, D. A., Garyantes, T., Green, D. V. S., Hertzberg, R. P., Janzen, W. P., Paslay, J. W., Schopfer, U., and Sittampalam, G. S. (2011). Impact of High-Throughput Screening in Biomedical Research. *Nature Reviews Drug Discovery 10*, 188–195, DOI: 10.1038/nrd3368.

(7) Santanilla, A. B., Regalado, E. L., Pereira, T., Shevlin, M., Bateman, K., Campeau, L.-C., Schneeweis, J., Berritt, S., Shi, Z.-C., Nantermet, P., Liu, Y., Helmy, R., Welch, C. J., Vachal, P., Davies, I. W., Cernak, T., and Dreher, S. D. (2015). Nanomole-Scale High-Throughput Chemistry for the Synthesis of Complex Molecules. *Science 347*, 49–53, DOI: 10.1126/science.1259203.

(8) MacLeod, B. P., Parlane, F. G. L., Morrissey, T. D., Häse, F., Roch, L. M., Dettelbach, K. E., Moreira, R., Yunker, L. P. E., Rooney, M. B., Deeth, J. R., Lai, V., Ng, G. J., Situ, H., Zhang, R. H., Elliott, M. S., Haley, T. H., Dvorak, D. J., Aspuru-Guzik, A., Hein, J. E., and Berlinguette, C. P. (2020). Self-driving laboratory for accelerated discovery of thin-film materials. *Science Advances 6*, Publisher: American Association for the Advancement of Science, eaaz8867, DOI: 10.1126/sciadv.aaz8867.

(9) 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., and Takeuchi, I. (2020). On-The-Fly Closed-Loop Materials Discovery Via Bayesian

Active Learning. *Nature Communications 11*, 5966, DOI: 10.1038/s41467-020-19597-w.

(10) Jain, A., Ong, S. P., Chen, W., Medasani, B., Qu, X., Kocher, M., Brafman, M., Petretto, G., Rignanese, G.-M., Hautier, G., Gunter, D., and Persson, K. A. (2015). Fireworks: a Dynamic Workflow System Designed for High-throughput Applications. *Concurrency and Computation: Practice and Experience 27*, 5037–5059, DOI: 10.1002/cpe.3505.

(11) Mölder, F., Jablonski, K. P., Letcher, B., Hall, M. B., Tomkins-Tinch, C. H., Sochat, V., Forster, J., Lee, S., Twardziok, S. O., Kanitz, A., Wilm, A., Holtgrewe, M., Rahmann, S., Nahnsen, S., and Köster, J. (2021). Sustainable Data Analysis With Snakemake. *F1000Research 10*, 33, DOI: 10.12688/f1000research.29032.1.

(12) Huber, S. P. et al. (2020). Aiida 1.0, a Scalable Computational Infrastructure for Automated Reproducible Workflows and Data Provenance. *Scientific Data 7*, 300, DOI: 10.1038/s41597-020-00638-4.

(13) Bo, C., Maseras, F., and López, N. (2018). The Role of Computational Results Databases in Accelerating the Discovery of Catalysts. *Nature Catalysis 1*, 809–810, DOI: 10.1038/s41929-018-0176-4.

(14) Stach, E. et al. (2021). Autonomous Experimentation Systems for Materials Development: a Community Perspective. *Matter 4*, 2702–2726, DOI: 10.1016/j.matt.2021.06.036.

(15) Abdel-Latif, K., Epps, R. W., Bateni, F., Han, S., Reyes, K. G., and Abolhasani, M. (2021). Self-Driven Multistep Quantum Dot Synthesis

Enabled by Autonomous Robotic Experimentation in Flow. *Advanced Intelligent Systems 3*, 2000245, DOI: 10.1002/aisy.202000245.

(16) Tao, H., Wu, T., Kheiri, S., Aldeghi, M., Aspuru-Guzik, A., and Kumacheva, E. (2021). Self-Driving Platform for Metal Nanoparticle Synthesis: Combining Microfluidics and Machine Learning. *Advanced Functional Materials 31*, 2106725, DOI: 10.1002/adfm.202106725.

(17) Li, J., Li, J., Liu, R., Tu, Y., Li, Y., Cheng, J., He, T., and Zhu, X. (2020). Autonomous discovery of optically active chiral inorganic perovskite nanocrystals through an intelligent cloud lab. *Nature Communications 11*, Number: 1 Publisher: Nature Publishing Group, 2046, DOI: 10.1038/s41467-020-15728-5.

(18) Roch, L. M., Häse, F., Kreisbeck, C., Tamayo-Mendoza, T., Yunker, L. P. E., Hein, J. E., and Aspuru-Guzik, A. (2020). ChemOS: An orchestration software to democratize autonomous discovery. *PLOS ONE 15*, Publisher: Public Library of Science, e0229862, DOI: 10.1371/journal.pone.0229862.

(19) Deneault, J. R., Chang, J., Myung, J., Hooper, D., Armstrong, A., Pitt, M., and Maruyama, B. (2021). Toward Autonomous Additive Manufacturing: Bayesian Optimization on a 3d Printer. *MRS Bulletin 46*, 566–575, DOI: 10.1557/s43577-021-00051-1.

(20) Tamura, R., Tsuda, K., and Matsuda, S. (2023). Nims-Os: an Automation Software To Implement a Closed Loop Between Artificial Intelligence and Robotic Experiments in Materials Science. *CoRR*.

(21) Kusne, A. G., and McDannald, A. (2023). Scalable Multi-Agent Lab Framework for Lab Optimization. *Matter 6*, 1880–1893, DOI: 10.1016/j.matt.2023.03.022.

(22) Rahmanian, F., Flowers, J., Guevarra, D., Richter, M., Fichtner, M., Donnely, P., Gregoire, J. M., and Stein, H. S. (2022). Enabling Modular Autonomous Feedback-loops in Materials Science Through Hierarchical Experimental Laboratory Automation and Orchestration. *Advanced Materials Interfaces 9*, 2101987, DOI: 10.1002/admi.202101987.

(23) Maffettone, P. M., Campbell, S., Hanwell, M. D., Wilkins, S., and Olds, D. (2022). Delivering Real-Time Multi-Modal Materials Analysis With Enterprise Beamlines. *Cell Reports Physical Science 3*, 101112, DOI: 10.1016/j.xcrp.2022.101112.

(24) Fitzpatrick, D. E., Maujean, T., Evans, A. C., and Ley, S. V. Across-the-World Automated Optimization and Continuous-Flow Synthesis of Pharmaceutical Agents Operating Through a Cloud-Based Server - Fitzpatrick - 2018 - Angewandte Chemie International Edition - Wiley Online Library.

(25) Fitzpatrick, D. E., Battilocchio, C., and Ley, S. V. (2016). A Novel Internet-Based Reaction Monitoring, Control and Autonomous Self-Optimization Platform for Chemical Synthesis. *Organic Process Research & Development 20*, Publisher: American Chemical Society, 386–394, DOI: 10.1021/acs.oprd.5b00313.

(26) Leong, C. J., Low, K. Y. A., Recatala-Gomez, J., Velasco, P. Q., Vissol-Gaudin, E., Tan, J. D., Ramalingam, B., Made, R. I., Pethe, S. D., Sebastian, S., Lim, Y.-F., Khoo, Z. H. J., Bai, Y., Cheng, J. J. W., and Hippalgaonkar, K. (2022). An object-oriented framework to enable workflow evolution across materials acceleration platforms. *Matter 5*, Publisher: Elsevier, 3124–3134, DOI: 10.1016/j.matt.2022.08.017.

(27) Bromig, L., Leiter, D., Mardale, A.-V., von den Eichen, N., Bieringer, E., and Weuster-Botz, D. (2022). The SiLA 2 Manager for rapid device integration and workflow automation. *SoftwareX 17*, 100991, DOI: 10.1016/j.softx.2022.100991.

(28) Manzano, J. S., Hou, W., Zalesskiy, S. S., Frei, P., Wang, H., Kitson, P. J., and Cronin, L. (2022). An autonomous portable platform for universal chemical synthesis. *Nature Chemistry 14*, Number: 11 Publisher: Nature Publishing Group, 1311–1318, DOI: 10.1038/s41557-022-01016-w.

(29) Rohrbach, S., Šiaučiulis, M., Chisholm, G., Pirvan, P.-A., Saleeb, M., Mehr, S. H. M., Trushina, E., Leonov, A. I., Keenan, G., Khan, A., Hammer, A., and Cronin, L. (2022). Digitization and validation of a chemical synthesis literature database in the ChemPU. *Science 377*, Publisher: American Association for the Advancement of Science, 172–180, DOI: 10.1126/science.abo0058.

(30) Fakhruldeen, H., Pizzuto, G., Glowacki, J., and Cooper, A. I. In *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp 6013–6019, DOI: 10.1109/ICRA46639.2022.9811996.

(31) Steiner, S., Wolf, J., Glatzel, S., Andreou, A., Granda, J. M., Keenan, G., Hinkley, T., Aragon-Camarasa, G., KItson, P., Angelone, D., and Cronin, L. Organic synthesis in a modular robotic system driven by a chemical programming language | Science.

(32) Shi, W., Cao, J., Zhang, Q., Li, Y., and Xu, L. (2016). Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal 3*, 637–646.

(33) Yi, S., Hao, Z., Qin, Z., and Li, Q. In *2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, 2015, pp 73–78, DOI: 10.1109/HotWeb.2015.22.

(34) Ritchie, D. M., and Thompson, K. (1974). The Unix Time-Sharing System. *Communications of the ACM 17*, 365–375, DOI: 10.1145/361011.361061.

(35) Raymond, E. S., *The art of UNIX programming*; Addison-Wesley professional computing series; Addison-Wesley Educational: Boston, MA, 2003.

(36) Wilkinson, M. D. et al. (2016). The Fair Guiding Principles for Scientific Data Management and Stewardship. *Scientific Data 3*, 160018, DOI: 10.1038/sdata.2016.18.

(37) Dolstra, E., and Löh, A. In *Proceedings of the 13th ACM SIGPLAN international conference on Functional programming*, Association for Computing Machinery: New York, NY, USA, 2008, pp 367–378, DOI: 10.1145/1411204.1411255.

(38) Sim, M. M. ChemOS 2.0, https://github.com/malcolmsimgithub/ChemOS2.0.

(39) Álvarez-Moreno, M., de Graaf, C., López, N., Maseras, F., Poblet, J. M., and Bo, C. (2014). Managing the Computational Chemistry Big Data Problem: the Iochem-Bd Platform. *Journal of Chemical Information and Modeling 55*, 95–103, DOI: 10.1021/ci500593j.

(40) Berman, H. M. (2000). The Protein Data Bank. *Nucleic Acids Research 28*, 235–242, DOI: 10.1093/nar/28.1.235.

(41) Jain, A., Ong, S. P., Hautier, G., Chen, W., Richards, W. D., Dacek, S., Cholia, S., Gunter, D., Skinner, D., Ceder, G., and Persson, K. A. (2013). Commentary: the Materials Project: a Materials Genome Approach To Accelerating Materials Innovation. *APL Materials 1*, 011002, DOI: 10.1063/1.4812323.

(42) Chanussot, L., Das, A., Goyal, S., Lavril, T., Shuaibi, M., Riviere, M., Tran, K., Heras-Domingo, J., Ho, C., Hu, W., et al. (2021). Open catalyst 2020 (OC20) dataset and community challenges. *Acs Catalysis 11*, 6059–6072.

(43) Miret, S., Lee, K. L. K., Gonzales, C., Nassar, M., and Spellings, M. (2023). The Open MatSci ML Toolkit: A Flexible Framework for Machine Learning in Materials Science. *Transactions on Machine Learning Research*.

(44) Li, J., Ballmer, S. G., Gillis, E. P., Fujii, S., Schmidt, M. J., Palazzolo, A. M. E., Lehmann, J. W., Morehouse, G. F., and Burke, M. D. (2015). Synthesis of many different types of organic small molecules using one automated process. *Science*, Publisher: American Association for the Advancement of Science, DOI: 10.1126/science.aaa5414.

(45) Seifrid, M., Pollice, R., Aguilar-Granda, A., Morgan Chan, Z., Hotta, K., Ser, C. T., Vestfrid, J., Wu, T. C., and Aspuru-Guzik, A. (2022). Autonomous Chemical Experiments: Challenges and Perspectives on Establishing a Self-Driving Lab. *Accounts of Chemical Research 55*, Publisher: American Chemical Society, 2454–2466, DOI: 10.1021/acs.accounts.2c00220.

(46) Wu, T. C., Granda, A. A., Hotta, K., Yazdani, S. A., Pollice, R., Vestfrid, J., Hao, H., Lavigne, C., Seifrid, M., Angello, N., Bencheikh, F., Hein, J. E., Burke, M., Adachi, C., and Aspuru-Guzik, A. A Materials Acceleration Platform for Organic Laser Discovery, 2022, DOI: 10.26434/chemrxiv-2022-9zm65.

(47) Wu, T. C., Aguilar-Granda, A., Hotta, K., Yazdani, S. A., Pollice, R., Vestfrid, J., Hao, H., Lavigne, C., Seifrid, M., Angello, N., Bencheikh, F., Hein, J. E., Burke, M., Adachi, C., and Aspuru-Guzik, A. (2023). A Materials Acceleration Platform for Organic Laser Discovery. *Advanced Materials 35*, 2207070, DOI: 10.1002/adma.202207070.

(48) O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. (2011). Open Babel: an Open Chemical Toolbox. *Journal of Cheminformatics 3*, 33, DOI: 10.1186/1758-2946-3-33.

(49) Pracht, P., Bohle, F., and Grimme, S. (2020). Automated Exploration of the Low-Energy Chemical Space With Fast Quantum Chemical Methods. *Physical Chemistry Chemical Physics 22*, 7169–7192, DOI: 10.1039/c9cp06869d.

(50) Bannwarth, C., Caldeweyher, E., Ehlert, S., Hansen, A., Pracht, P., Seibert, J., Spicher, S., and Grimme, S. (2020). Extended <scp>tight-binding</scp> Quantum Chemistry Methods. *WIREs Computational Molecular Science 11*, nil, DOI: 10.1002/wcms.1493.

(51) Neese, F., Wennmohs, F., Becker, U., and Riplinger, C. (2020). The Orca Quantum Chemistry Program Package. *The Journal of Chemical Physics 152*, 224108, DOI: 10.1063/5.0004608.

(52) de Souza, B., Neese, F., and Izsák, R. (2018). On the Theoretical Prediction of Fluorescence Rates From First Principles Using the Path Integral Approach. *The Journal of Chemical Physics 148*, 034104, DOI: 10.1063/1.5010895.

(53) Snowflake streamlit https://streamlit.io (accessed July 11, 2023).

(54) Hickman, R. J., Sim, M., Pablo-García, S., Woolhouse, I., Hao, H., Bao, Z., Bannigan, P., Allen, C., Aldeghi, M., and Aspuru-Guzik, A. Atlas: A Brain for Self-driving Laboratories, unpublished, https://github.com/aspuru-guzik-group/atlas.

(55) Griffiths, R.-R. et al. GAUCHE: A Library for Gaussian Processes in Chemistry, arXiv:2212.04450 [cond-mat, physics:physics], 2023, DOI: 10.48550/arXiv.2212.04450.

(56) Angello, N. H., Rathore, V., Beker, W., Wołos, A., Jira, E. R., Roszak, R., Wu, T. C., Schroeder, C. M., Aspuru-Guzik, A., Grzybowski, B. A., and Burke, M. D. (2022). Closed-loop optimization of general reaction conditions for heteroaryl Suzuki-Miyaura coupling. *Science 378*, Pub-

lisher: American Association for the Advancement of Science, 399–405, DOI: 10.1126/science.adc8743.

(57) Aldeghi, M., Häse, F., Hickman, R. J., Tamblyn, I., and Aspuru-Guzik, A. (2021). Golem: an algorithm for robust experiment and process optimization. *Chemical Science*, Publisher: The Royal Society of Chemistry, DOI: 10.1039/D1SC01545A.

(58) Ginsbourger, D., Janusevskis, J., and Riche, R. L. (2011). Dealing with asynchronicity in parallel Gaussian Process based global optimization.

(59) Desautels, T., Krause, A., and Burdick, J. W. (2014). Parallelizing Exploration-Exploitation Tradeoffs in Gaussian Process Bandit Optimization. *Journal of Machine Learning Research 15*, 4053–4103.

(60) Rasmussen, C. E., and Williams, C. K. I., *Gaussian processes for machine learning*. Adaptive computation and machine learning; MIT Press: 2006, pp I–XVIII, 1–248.

(61) MacLeod, B. P., Parlane, F. G. L., Brown, A. K., Hein, J. E., and Berlinguette, C. P. (2022). Flexible automation accelerates materials discovery. *Nature Materials 21*, Number: 7 Publisher: Nature Publishing Group, 722–726, DOI: 10.1038/s41563-021-01156-3.

(62) Bran, A. M., Cox, S., White, A. D., and Schwaller, P. (2023). Chemcrow: Augmenting Large-Language Models With Chemistry Tools. *CoRR*.

(63) Oyama, Y., Mamada, M., Shukla, A., Moore, E. G., Lo, S.-C., Namdas, E. B., and Adachi, C. (2020). Design Strategy for Robust Organic Semiconductor Laser Dyes. *ACS Materials Letters*, Publisher: American Chemical Society, DOI: 10.1021/acsmaterialslett.9b00536.

(64) Bayer, M. In *The Architecture of Open Source Applications Volume II: Structure, Scale, and a Few More Fearless Hacks*, Brown, A., and Wilson, G., Eds.; aosabook.org: 2012.

(65) Psycopg2 – PostgreSQL database adapter for Python — Psycopg 2.9.6 documentation https://www.psycopg.org/docs/ (accessed July 5, 2023).

(66) Häse, F., Aldeghi, M., Hickman, R. J., Roch, L. M., Christensen, M., Liles, E., Hein, J. E., and Aspuru-Guzik, A. (2021). Olympus: a benchmarking framework for noisy optimization and experiment planning. *Machine Learning: Science and Technology 2*, Publisher: IOP Publishing, 035021, DOI: 10.1088/2632-2153/abedc8.

(67) Huber, S. P. aiida-shell, version 0.5.3, École Polytechnique Fédérale de Lausanne, 2023.

(68) Pablo-García, S., and Sim, M. ChemOS 2.0 DFT database, https://www.iochem-bd.org/handle/10/346190.