ORGANA: A Robotic Assistant for Automated Chemistry Experimentation and Characterization

Kourosh Darvish[1,2†*], Marta Skreta[1,2†], Yuchi Zhao[1†], Naruki Yoshikawa[1,2], Sagnik Som[1], Miroslav Bogdanovic[1], Yang Cao[1], Han Hao[1], Haoping Xu[1,2], Alán Aspuru-Guzik[1,2‡], Animesh Garg[1,2,3‡], Florian Shkurti[1,2‡]

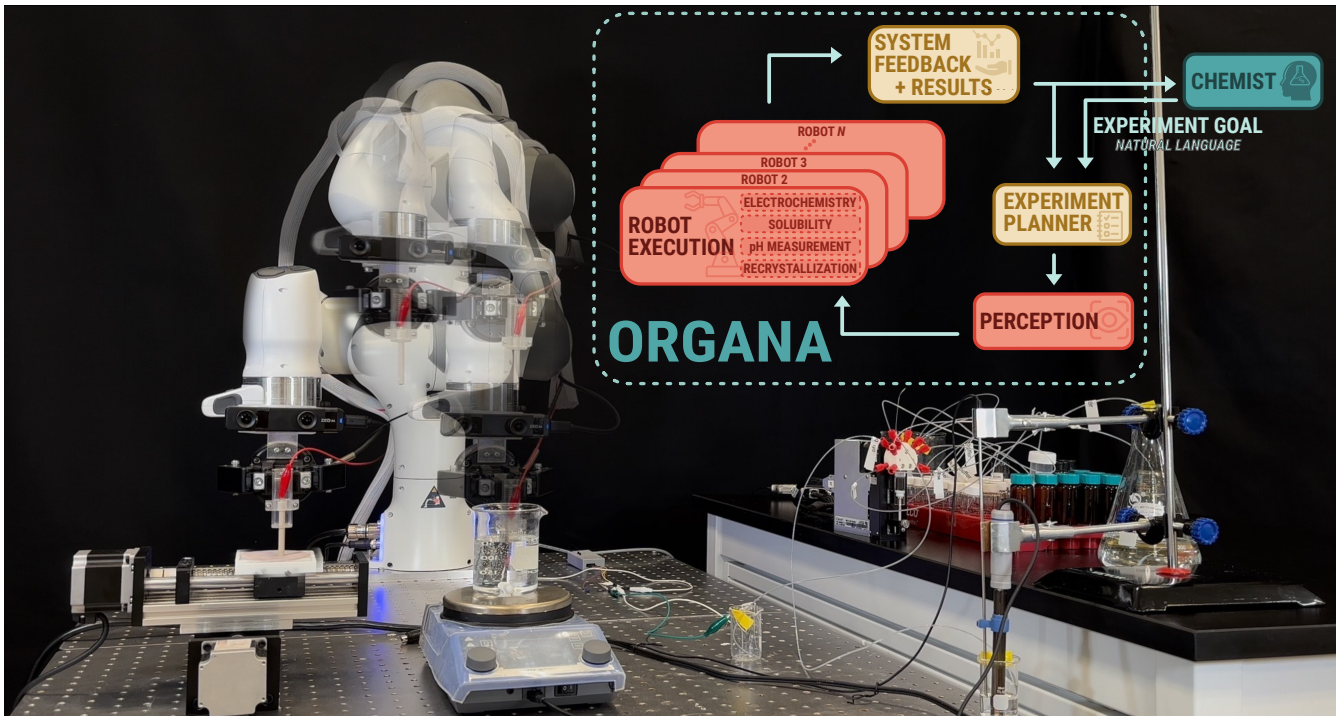


Fig. 1: **Robot setup with ORGANA's overall schema.** ORGANA provides a seamless interaction between SDLs and chemists for diverse chemistry experiments. It perceives the objects and the progress of the chemistry task to make an informed decision for the next steps of the chemistry experiment. Informed decisions, guided by user intention and perception, are a key strength. ORGANA optimizes SDL efficiency through parallel experiment execution, providing timely feedback via reports and analysis, thus keeping users well-informed and involved in high-level decision-making. More details about ORGANA can be found at https://ac-rad.github.io/organa/.

***Abstract*— Chemistry experimentation is often resource- and labor-intensive. Despite the many benefits incurred by the integration of advanced and special-purpose lab equipment, many aspects of experimentation are still manually conducted by chemists, for example, polishing an electrode in electrochemistry experiments. Traditional lab automation infrastructure faces challenges when it comes to flexibly adapting to new chemistry experiments. To address this issue, we propose a human-friendly and flexible robotic system, ORGANA, that automates a diverse set of chemistry experiments. It is capable of interacting with chemists in the lab through natural language, using Large Language Models (LLMs). ORGANA keeps scientists informed by providing timely reports that incorporate statistical analyses. Additionally, it actively engages with users when necessary for disambiguation or troubleshooting. ORGANA can reason over user input to derive experiment goals, and plan long sequences of both high-level tasks and low-level robot actions while using feedback from the visual perception of the environment. It also supports scheduling and parallel execution for experiments that require resource allocation and coordination between multiple robots and experiment stations. We show that ORGANA successfully conducts a diverse set of chemistry experiments, including solubility assessment, pH measurement, recrystallization, and electrochemistry experiments. For the latter, we show that ORGANA robustly executes a long-horizon plan, comprising 19 steps executed in parallel, to characterize the electrochemical properties of quinone derivatives, a class of**

[1]University of Toronto, Toronto, ON, Canada
[2]Vector Institute, Toronto, ON, Canada
[3]NVIDIA, Santa Clara, CA, USA
[†]These authors equally contributed to this work.
[‡]These authors equally supervised this work.
[*]Corresponding author; E-mail: kdarvish@cs.toronto.edu

molecules used in rechargeable flow batteries. Our user study indicates that ORGANA significantly improves many aspects of user experience while reducing their physical workload.

Summary: We present ORGANA: a modular and human-friendly robotic system to enable automation in chemistry labs.

I. INTRODUCTION

The process of discovering materials, from generating candidates to conducting experiments, is time- and labor-intensive. It entails synthesizing and characterizing samples at various scales (ranging from milligrams to grams) in order to identify the desired material that meets the specified requirements. While chemistry labs use a wide variety of special-purpose equipment to expedite independent tasks including synthesis, purification, and analysis, their specialized nature hinders flexible and modular automation that supports a diverse set of tasks. Achieving fully automated Self-Driving Labs (SDLs), with data-driven experiment planning and automated experiment execution, to accelerate material discovery in the vast chemical search space remains a formidable challenge [1, 2]. This involves integrating general-purpose robots with lab equipment to efficiently perform chemistry tasks, perceive the environment, plan actions, facilitate high-throughput experiments, interact intuitively with chemists, and maintain safety while adapting flexibly to new chemistry tasks. Furthermore, automatic experimentation in SDLs can enhance result consistency and reproducibility, crucial for scientific discoveries. Such systems could improve the accessibility of chemistry experiments to users, especially in scenarios of dangerous operations and users with disabilities.

There have been various demonstrations of automated labs in the literature for material discovery [3–5]; however, a pivotal question that remains unaddressed is how to enable efficient and user-friendly automated labs capable of adapting flexibly to lab environments and supporting diverse chemistry tasks [6]. In this work, we introduce ORGANA, a flexible and user-friendly robotic solution designed for chemistry lab automation. ORGANA, as demonstrated in Figure 1, is empowered by LLMs to interact with chemists, identify their intentions, and plan robot experiments. It also offers feedback to chemists by analyzing chemistry experiment outputs. Its reasoning and intuitive interaction capabilities allow for effective communication between chemists and ORGANA, thereby reducing human effort and keeping chemists informed of high-level decisions unobtrusively. ORGANA provides chemists with a summary of experiments, their results, and analysis in the form of a report. This approach informs chemists with comprehensive feedback and enables timely user intervention when necessary [7]. Another aspect of ORGANA is its 3D visual perception capabilities, which enable manipulation of objects as well as monitor progress of chemistry experiments. This allows ORGANA to make informed, closed-loop decisions on how robots will interact with lab equipment and when to proceed to the next step of a long experiment. The combination of closed-loop decision-making and high-level human involvement when necessary contributes to the overall robustness of the system and can reduce downtime. Additionally, ORGANA is designed to be modular in terms of both hardware and functional components, empowering scientists to adopt them on demand for various purposes and a diverse set of experiments. To expedite material discovery, a key element is the ability to perform high-throughput experimentation by parallelizing multiple experiments or segments. ORGANA supports the parallel execution of chemistry experiments, reducing the overall makespan and enhancing efficiency.

Compared to our prior work [8], ORGANA can reason over human instructions and provide feedback to users about the experimental results in the form of reports. It engages the user in troubleshooting if unexpected behavior, outliers, or ambiguities have occurred during the execution of an experiment. These capabilities of ORGANA lead to more efficient interaction with the user, compared to [8]. Moreover, it pushes beyond previous work in terms of skills, for example, by the perception of transparent objects, which are essential for chemistry task execution. Finally, ORGANA makes chemistry experiments more efficient by allowing for the parallel execution of chemistry tasks, while [8] only supports sequential execution.

As shown in Figure 1, ORGANA receives commands from chemists in audio or text format, translates them using an LLM into a detailed chemistry task description, and then maps these instructions to the robot's goals. Additionally, it grounds perceived objects in the scene, obtained from perception, through user interaction. ORGANA improves efficiency by simultaneously solving task and motion planning (TAMP) and scheduling problems, enabling parallel execution of tasks. Moreover, ORGANA provides feedback to users by offering a comprehensive report and analysis and notifying them in case of unexpected results. Various instances of ORGANA are used to execute diverse chemistry experiments, such as solubility screening, recrystallization, and pH experimentation. ORGANA is utilized to identify the electrochemical characteristics of quinone, a promising molecule for redox flow batteries. The system is also evaluated by conducting a user study with chemists, validating its significant usefulness with effective communication. The summary of our contributions is as follows:

1) We introduce a modular and user-friendly robotic system designed to flexibly support a diverse set of chemistry experiments, capable of interacting with both standard and affordable custom-made lab equipment. Our system also includes automatic analysis and report generation, as illustrated in Figure 2.
2) We are the first to demonstrate the automation of characterizing the electrochemical properties of a quinone derivative (anthraquinone-2-sulfonate, AQS) through fully automated mechanical polishing, as shown in Figure 5. Automating mechanical polishing in electrochemistry is of significant practical importance to chemists.
3) We enhance efficiency in experimentation and utilization of lab resources by simultaneously solving TAMP and scheduling problems, enabling the parallel execu-

tion of chemistry tasks. The advantage is that chemists do not need to specify a custom finite-state machine for every new type of experiment they want to automate.

II. Related Work

a) Chemistry Lab Automation: Recent advances in lab automation have witnessed diverse efforts, with a focus on special-purpose hardware. For instance, the synthesis automation detailed in [3, 9] utilizes a liquid handler, formalizing chemical synthesis and connecting it to physical operations through a structured language. Another example is Electrolab, an electrochemistry system capable of autonomously formulating redox electrolytes and assessing them through cyclic voltammetry (CV) under a range of conditions without human input [10]. Efforts have been made to overcome the limitation of specialized hardware through the use of a robotic system [11]. For example, [4] employed general-purpose robots to demonstrate the use of a mobile manipulator to operate instruments designed for human chemists and conduct a specific chemistry experiment. An effort to enhance reconfigurability for chemistry experiments is demonstrated in [12], employing an intuitive set of state machines and workflows. While these initiatives were conducted in structured lab setups with known object poses, recent works, such as [5], introduce a modular architecture, leveraging QR codes for perception and enhancing flexibility. Commonly among them, the perception of chemical reactions and objects in the scene [13–15] introduce significant challenges to lab automation, especially given the transparency of chemistry lab tools [16, 17]. Another illustration [18] showcases the usability of robotic systems in realizing several novel compounds, executing experiments over an extended period with the option of human intervention when necessary. In another example, a graphical user interface (GUI) was used in [19] to allow users to encode electrochemistry experiments for robot execution using a fill-in-the-blank template. Despite these promising steps towards SDLs for accelerated material discovery, challenges persist in terms of flexibility, modularity, robustness, and human-centric automation across these examples. An inspiring example is found in [20], which envisions a system featuring a fictitious character, Organa, engaging in dialogue with a chemist and providing answers to any chemical question.

b) Language Models for Robotics in Lab Automation: Automation in chemistry labs poses a challenge for chemists unfamiliar with robotic programming, highlighting recent efforts to simplify interaction through natural language interfaces. Chemistry experiment translators have employed rule-based approaches [3] or fine-tuned sequence-to-sequence models for mapping to robot plans [21]. Mehr et al. introduced a hardware-agnostic Chemical Description Language (XDL) in XML format, facilitating integration into diverse robotics infrastructures [3].

Recently, Large Language Models (LLMs) are increasingly employed for planning chemistry experiments due to their reasoning capabilities [22]. CLAIRify, as an example, utilizes LLM to translate natural language experiments into XDL. The process involves iterative prompting of the LLM to ensure syntactic correctness, and the resulting plans can be executed on a Franka robot in a chemistry lab [8]. ChemCrow, an LLM-powered chemistry engine, integrates external tools for molecule synthesis planning, utilizing a list of 18 available tools, including calculators, literature search tools, and IBM's RoboRXN platform. It successfully demonstrated the synthesis of two real-world molecules from literature—an insect repellent and a Diels-Alder reaction catalyst [23]. Boike et al. designed an LLM-based Intelligence Agent capable of autonomously designing, planning, and conducting scientific experiments by performing tasks like browsing the internet, using liquid transferring robots, and selecting relevant functions from hardware documents [24]. Inspired by existing approaches integrating LLM-based models into experiment planning and robot execution, our focus is on monitoring reactions, detecting *unexpected outcomes*, and providing feedback to chemists for corrective actions. We explore LLMs generating automatic summary reports for chemists post-experiment, drawing inspiration from the Automatic Statistician project [7], which assesses models on datasets, providing summaries of performance, prediction explanations, and criticisms. Similarly, our approach generates reports summarizing robot plans, experiment results, data analysis plots, experiment errors, and human-provided information.

c) Task and Motion Planning and Scheduling: Typically, works in the lab automation literature encode robot and lab device plans using manual state machines[3, 4]; nevertheless, they face challenges in adapting to uncertainties in the environment. To autonomously solve the problem of robot executable plan generation and parallelizing the robot or agent actions, it is required to solve the problem of TAMP as well as the scheduling problem simultaneously. In the literature, TAMP problem has been addressed with different techniques. An approach to solve TAMP problem is PDDLStream [25], which combines Planning Domain Definition Language (PDDL) solvers for solving discrete planning probelms [26] with *streams*, a declarative sampling procedure. In another work for geometry-rich sequential robot manipulation problem [27], the authors solve it as an optimization problem. In an extension, [28] solves tool-use problems with mixed-integer programming combined with kinematic and differentiable dynamics constraints. Also, learning-based approaches were proposed in the literature [29–31] to address the efficiency shortcomings of previous methods.

High-throughput experimentation requires the parallel execution of chemistry tasks, a challenge addressed through scheduling methods discussed in the literature [32, 33]. Scheduling typically deals with a fixed sequence of actions and is commonly resolved using optimization techniques such as mixed-integer linear programming [34]. In this work, we address TAMP and scheduling problems simultaneously. Simultaneous task planning and scheduling, also known as temporal planning, deals with simultaneous durative actions during planning, often utilizing PDDL2.1 [34, 35]. While TAMP and scheduling problems have been explored indi-

vidually in the literature, their integration has not received sufficient attention. Two examples of such combinations are presented in [36], where temporal planning and a sampling-based motion planner solve the problem in two steps, and in [37], where mixed-integer linear programming was employed. Furthermore, while [25] provides a simple example code, it lacks a formal description for addressing simultaneous scheduling and TAMP.

III. Methods

The architecture, workflow, and main components of ORGANA are described in Figure 2. The following sections elaborate on the details of each component.

A. Large Language Model (LLM)-Based Interaction and Reasoning

Generating low-level robot plans for each experiment can be burdensome for chemists. To streamline this, ORGANA.REASONER facilitates the process through three steps: i) generating a natural language chemistry experiment task description from the intention of chemists for an experiment through an interactive conversation, ii) translating the experiment's natural description into valid structured language, and iii) resolving ambiguities by interacting with the user during experiments, including grounding perception information and addressing unexpected outcomes.

a) Autonomous experiment reasoning: ORGANA.REASONER autonomously generates several experiments' natural descriptions by identifying user intentions through a startup phase interactive conversation. This includes inquiries about experiment goals, expected observations, rationale, sample experiment procedures, and available reagents in the scene. Leveraging a summary of observations from past experiments and the current time step, ORGANA.REASONER employs the ReAct prompting scheme [38] to generate experimental plans using *(thought, action, observation)* tuples. An *action* is viewed as an experimental plan, *observations* are the measured experimental values post-plan execution, and *thought* represents the rationale behind a given experiment. Details about user interaction modalities (text or speech) and implementation specifics can be found in Appendix VIII-B.

b) Experiment description to valid structured task: We employ CLAIRify [39] to convert the natural language description of a chemistry experiment into structured language codes in the XDL language, which are used as goals for planning by ORGANA.PLANNER. CLAIRify utilizes an iterative prompting scheme to guarantee syntactic validity in the output language domain.

c) Human-in-the-loop disambiguation and troubleshooting: ORGANA engages with the user beyond experiment planning for scene clarification and resolving inconsistencies between *observations* and *expected observations*. Following [40], ORGANA addresses scene ambiguities during startup by grounding object functionalities. To handle unexpected experimental outcomes, a human-in-the-loop approach is adopted, where ORGANA.REASONER reasons over *observations* and *expected observations*, prompting the user to investigate and decide on further actions. Examples of ambiguity and uncertainty resolution and their prompts are detailed in Appendix VIII-B.

B. Task and Motion Planning with Scheduling

To enable high-throughput experimentation, the TAMP planner should facilitate parallel task execution by robots and other resources or equipment. We adapted PDDLStream [25] with PDDL2.1 [35] to support durative actions and introduced a time-variant cost function to enhance task execution efficiency. PDDLStream is represented by the tuple $< \mathcal{P}, \mathcal{A}, \mathcal{I}, \mathcal{G}, \mathcal{S} >$, respectively defining predicates, actions, initial state, goal state, and streams. The stream $S(\mathbf{x})$ over a tuple of literals $\mathbf{x}$ acts as a conditional sampler, declaring the satisfaction of the relation between its input and output tuples. To enable durative actions, $\mathtt{a} \in \mathcal{A}$ is substituted with *starting* and *ending* actions, denoted as `a-start` and `a-end` [35]. Additionally, the starting time of the i'th action in the plan, where $a_i \in \pi$, is linked to $t_{\mathtt{a\text{-}start},i}$, and its duration is indicated by $D_{\mathtt{a},i}$. For efficiency and reduced task execution time, the total cost is defined as:

$$\mathtt{total\text{-}cost} = \sum_{i=1}^{k} (t_{\mathtt{a\text{-}start},i} + D_{\mathtt{a},i}) \quad \forall \mathtt{a} \in \pi, k = |\pi|. \tag{1}$$

Figure 3 details the transformation of action `a` to `a-start` and `a-end`, along with updates to their cost functions.

Algorithm 1 TEMPORAL-PDDLSTREAM

Input: $\mathcal{A}, \mathcal{S}^o, \mathcal{S}^c, \mathcal{I}, \mathcal{G}$
Output: π
$\mathcal{U}^c = \text{APPLYSTREAMS}(\mathcal{S}^c, \mathcal{I}, \boldsymbol{next})$ ▷ eagerly evaluate costs
while $True$ **do**
  $\mathcal{U}^* = \text{APPLYSTREAMS}(\mathcal{S}^o, \{\mathcal{I}, \mathcal{U}^c\}, \text{OPTOUTPUT})$ ▷ optimistic stream
  $\pi^* = \text{SEARCH}(\mathcal{A}, \{\mathcal{I}, \mathcal{U}^c, \mathcal{U}^*\}, \mathcal{G})$
  $\pi, \psi = \text{EVALUATE}(\{\mathcal{I}, \mathcal{U}^c\}, \mathcal{U}^*, \pi^*, \mathcal{G})$
  if $\pi \neq$ None **then return** π

In [25], various methods to solve the PDDLStream problem are discussed, including an incremental approach where streams are certified eagerly and blindly before the search, leading to inefficiency due to the generation of irrelevant facts during stream evaluation. Another method involves optimistic certification of streams, enabling a lazy exploration of candidate plans. While more efficient, this approach does not support time-varying functions with streams. In Temporal TAMP, costs, varying with time, are updated through streams. To overcome the limitations of the two existing methods in solving the temporal TAMP problem, we integrate both approaches. Specifically, time-varying streams linked to cost functions and timings are evaluated eagerly, while the remaining streams are evaluated optimistically. Additionally, by imposing reasonable constraints on eagerly evaluated streams, we restrict the search space to enhance search efficiency. Algorithm 1 outlines our method for solving temporal PDDLStream problems, taking input durative actions $\mathcal{A}$,

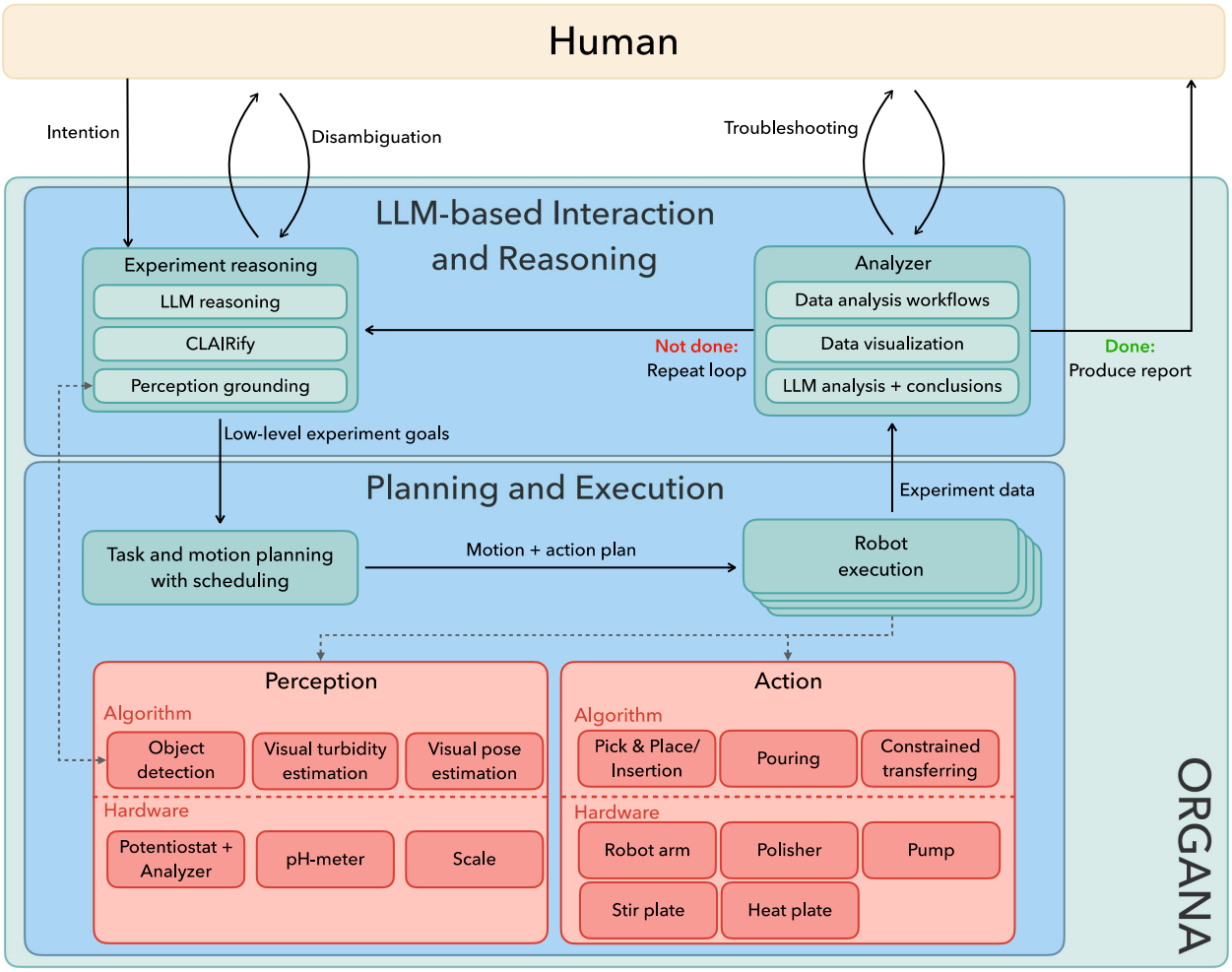


Fig. 2: **ORGANA's architecture and workflow.** Users interact with ORGANA to convey their intentions for chemistry experiments and disambiguate the functionality of objects by grounding the scene. The LLM-based ORGANA.REASONER module translates these instructions into chemistry experiment plans and goals. Subsequently, ORGANA.PLANNER generates parallel task and motion plans for execution, optimizing hardware utilization. ORGANA.ROBOTEXECUTION, equipped with action and perception skills, executes plans in parallel to maximize equipment usage and conduct experiments. The ORGANA.ANALYZER processes raw data to estimate experiment progress and provides feedback to determine the next steps. Human notifications occur only when necessary to address unexpected situations.

optimistic streams $\mathcal{S}^o$, cost-related eager streams $\mathcal{S}^c$, and initial states $\mathcal{I}$ with a goal $\mathcal{G}$. Initially, we assess time- and cost-associated streams, incorporating them into the current set of certified facts $\mathcal{U}^c$. The remaining streams $\mathcal{S}^o$ are optimistically evaluated using the `OptOutput` procedure to create an optimistic object tuple $u^* \in \mathcal{U}^*$. In line 4, we employ a fast downward planning system utilizing weighted A* heuristic search [41] to find an optimistic plan π^* with a focus on minimizing the plan cost. Finally, the optimistic plan and its associated streams are evaluated and certified, returning upon finding a plan.

C. Perception

To achieve autonomous chemistry experiments, we suggest a dual-level perception framework. The first level monitors chemical task progress by characterizing materials, while the second level focuses on perceiving workspace objects for robot manipulation. Our approach integrates various sensors and utilizes perception algorithms for monitoring reactions and estimating workspace states. Details of perception algorithms are provided below, and hardware specifics can be found in Appendix VIII-D.

a) Turbidity visual feedback: Turbidity, indicating solution opaqueness, gauges undissolved solvent in solubility experiments. Following HeinSight [15], we adopt the solution's average brightness, observed by a robot with an in-hand camera, as a proxy for turbidity. Using the Hough Circle Transform [42], the robot identifies the dish in a top-down view, extracts the minimum square region enclosing it in HSV color space, and computes the average brightness as the turbidity value. See Figure 4 for an automated turbidity measurement example.

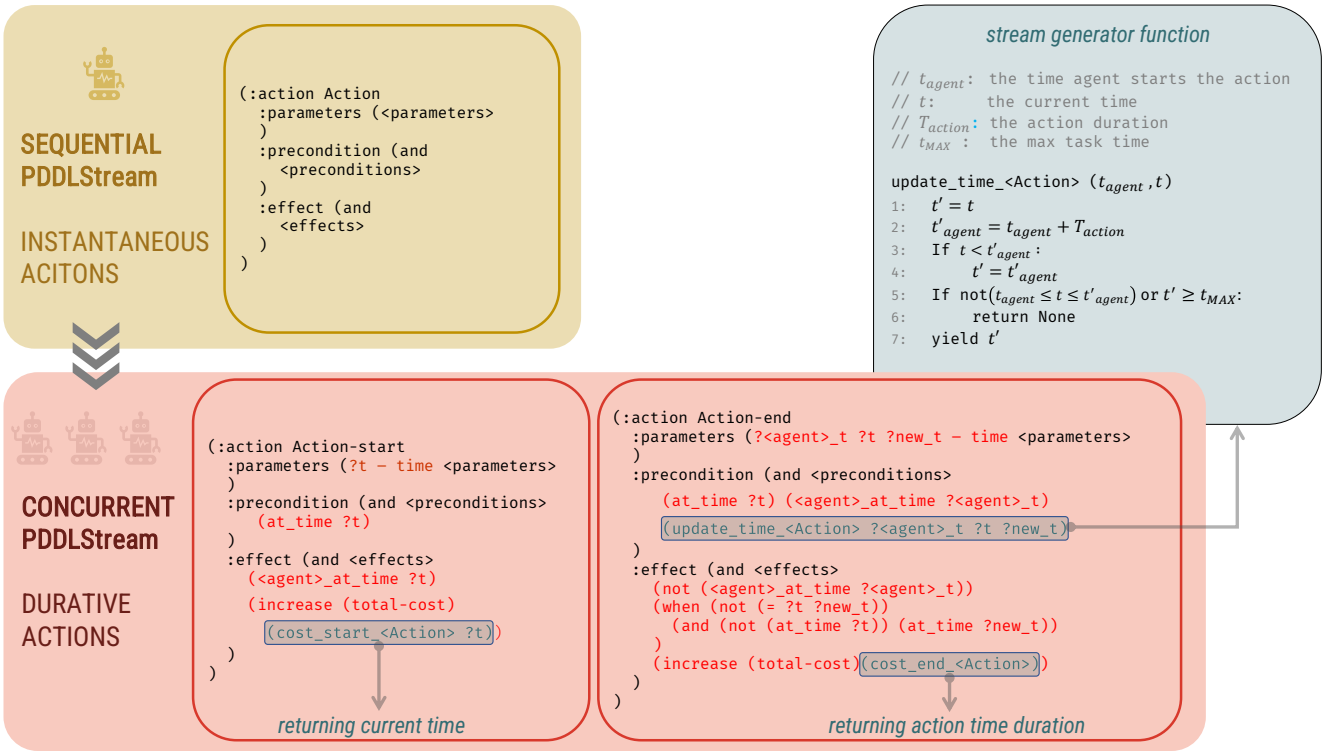


Fig. 3: **Using PDDLStream to solve task and motion planning problems with scheduling.** To support parallel task execution in ORGANA, we transform the instantaneous actions `a` $\in \mathcal{A}$ to durative actions from PDDL2.1 with starting `a-start` and ending `a-end` [35]. For this purpose, the preconditions and effects are updated accordingly to meet the requirements and constraints of concurrent plans. For example, while the agent is acting, it cannot be assigned to any other actions. In addition, a non-negative `(time ?t)` literal and a new predicate are added to keep track of the current execution time `(at_time ?t)` and the starting time of each action `(<agent>_at_time ?t)`. Two stream functions are added to provide the cost of action starting and ending. Moreover, `(update_time_<Action> ?<agent>_t ?t ?new_t)` is associated with a stream generator that gets as input starting time of the action `(?<agent>_t)` and the current execution time `(?t)` and it updates the current execution time with `(?new_t)`. The constraints on streams are added in line 5 of the stream generator function, ensuring the current time is higher than the action starting time, and the action is ended with the correct duration. Moreover, it ensures the updated time is less than the maximum time allowed to reach the goal of PDDLStream.

b) Transparent and opaque object detection and pose estimation: Perceiving transparent objects poses challenges due to violations of the Lambertian assumption and the textureless nature of transparent surfaces[43]. To address this, we integrated cutting-edge perception algorithms in ORGANA.PERCEPTION for effective transparent object detection and pose estimation in the scene (refer to Appendix VIII-A for the perception pipeline figure). Firstly, we employed GroundDINO [44], a transformer-based zero-shot object detection model [45] with grounded pre-training. This model enables flexible object detection by accepting human category names and images as input and providing bounding box information, labels, and confidence for each detection. Next, we applied Non-Maximum Suppression (NMS), a standard post-processing method in object detection [46], to remove duplicate detections and retain the most pertinent bounding boxes. The resulting bounding box was then fed into the Segment Anything model (SAM) [47], utilizing a transformer-based image encoder and mask decoder for segmentation. SAM offers options for points, boxes, and segment prompting through a CLIP-based prompt encoder [48]. Further implementation details can be found in Appendix VIII-A.

Estimating 3D object poses from 2D segments requires depth information for reprojection. We obtained the necessary depth data using the ZED camera depth map [49]. Opaque object point clouds maintain high accuracy, but transparency introduces distortion, capturing background surfaces. To mitigate this, a practical solution involves estimating the camera distance to the front surface of transparent objects by filtering the least distorted closest 10% of points (see Appendix VIII-A). The minimal 3D bounding box was derived from each object's point cloud convex hull using Open3D functions [50]. Principal Component Analysis (PCA) [51] identified the object's major axes, forming the rotation matrix of the object's frame of reference. Finally,

object pose in the robot world frame was determined through extrinsic camera calibration.

D. Skills for Chemistry Experiment

To enable autonomous chemistry experiments, it is crucial to integrate a diverse set of robot skills and laboratory tools. Skills are attained through either specialized hardware (refer to Appendix VIII-E) or diverse algorithms, as detailed below.

a) Pick & place and insertion skills: ORGANA uses object pose information coming from perception to determine the robot end-effector's target frame. These frames are used for grasping, placement, and insertion. To enhance robustness and avoid collisions, a pre/post pose strategy is applied, such as pre-insertion, insertion, and post-insertion poses during object manipulation. The robot joint trajectory is computed using inverse kinematics solved [52] and the probabilistic roadmap (PRM*) path planning [53].

b) Constrained motion planning skill: In a chemistry lab, a common task involves transporting containers with liquids and powders. In experiments, especially those related to solubility and recrystallization, we introduced orientation constraints to prevent spillage when the robot transfers beakers. We utilized the PRM* sampling-based method for robot motion planning [54]. To incorporate k-dimensional path constraints $\mathcal{F}(q) : \mathcal{Q} \to \mathbb{R}^k$ in the configuration space $\mathcal{Q}$, we applied a projection-based method to identify configurations satisfying constraints during PRM* sampling [53]. When sampling a free configuration in PRM*, its projected value in the constrained configuration space is determined by iteratively minimizing $\mathcal{F}(q)$ using its Jacobian. For details on our implementation and evaluation, see [8].

c) Liquid and granular material pouring skill: In the chemistry lab, liquid and granular solid pouring is routine, with liquid transfer handled by a pump. Precise powder pouring is challenging and expensive with existing hardware solutions. Inspired by manual pouring skills, ORGANA implements a robotic pouring technique in solubility and recrystallization experiments. This skill incorporates weight feedback using a PD controller and a shaping function, taking the desired substance target value as input and providing the desired rotational velocity of the robot end-effector. Additional details are available in [8].

E. Automated Data Analysis and Report Generation

To enable ORGANA to generate comprehensive user reports, we integrated the following tools into ORGANA.ANALYZER.

1) Electrochemistry Parameter Estimation: In the electrochemistry experiment, we aim to characterize the relationship between the pH and the redox potential, which is the potential that drives the reduction or oxidation half-reaction of a compound measured against a standard reference half-cell [55]. We know that the relationship has three distinct regions of linear dependency, demarcated by pH values $\mathrm{pK_{a1}}$ and $\mathrm{pK_{a2}}$. We also know that the slope of the second region (from $\mathrm{p}K_{\mathrm{a1}}$ to $\mathrm{p}K_{\mathrm{a2}}$) is one-half of the slope of the first one (before $\mathrm{p}K_{\mathrm{a1}}$), while in the last region (after $\mathrm{p}K_{\mathrm{a2}}$) the redox potential does not change (slope is equal to zero) (Section IV-D). The model is therefore fully defined with 4 parameters: two inflection points ($\mathrm{p}K_{\mathrm{a1}}$ and $\mathrm{p}K_{\mathrm{a2}}$), a single slope variable (in our case k, the slope in region $[\mathrm{pK_{a1}}, \mathrm{pK_{a2}}]$) and one variable to define the redox potential offset (we take $\mathrm{E_{inf}}$, the value in the third static region).

To produce parameter values based on collected data we utilize a maximum likelihood estimate (MLE). In addition to that, we aim to produce an updated belief about the parameter values and the model line after each new data point has been sampled. We utilize this posterior over parameters to plot marginal distributions for each individual one (see Figure 6 and Figure 8 for examples). With an automated system, this is an important element in keeping the chemist aware of the current progress of the experiment, in order to catch any issues that the system might not automatically detect and in general to make any corrections necessary. Additionally, while utilizing an optimization algorithm to choose points to sample was not needed in the specific electrochemistry experimental setup, this output would allow for the simple application of out of the box optimization algorithms in the future. We give details of the method for estimating the posterior distribution in the Appendix VIII-F.

2) Electrochemistry Report Generation: ORGANA automatically generates a PDF summary report at the experiment's conclusion, offering a comprehensive overview of the chemistry experiment, including automatically-generated statistical analyses of the measurements, to the users. The report includes experiment details, logs of failures with corresponding resolutions, and summary plots analyzing the results. An example electrochemistry report is in Appendix VIII-G.

IV. RESULTS

We assess ORGANA for its reliability in reproducing literature results and modularity through a diverse set of multistep chemistry experiments, including solubility screening, recrystallization, pH testing, and electrochemistry. While the first three experiments were previously detailed in [8], they are briefly reiterated here for completeness, alongside results from a new electrochemistry experiment, where we show automation of the labor-intensive process of electrode polishing for the first time. Additionally, we evaluate the interaction between chemists and ORGANA through a user study.

A. Solubility Experiment

Solubility is the physical property describing the highest concentration of a solute that is capable of dissolving in a solvent under a specific temperature. For this experiment, the robot iteratively pours a small amount of water until all solids are dissolved. After each pouring, the solution is stirred, and the turbidity, a quantitative measurement of residue in solutions, is estimated via a vision-based algorithm that we adapted from [15]. In Figure 4, an instance of ORGANA is shown conducting a solubility experiment, with the robot observing the dissolved solution for turbidity

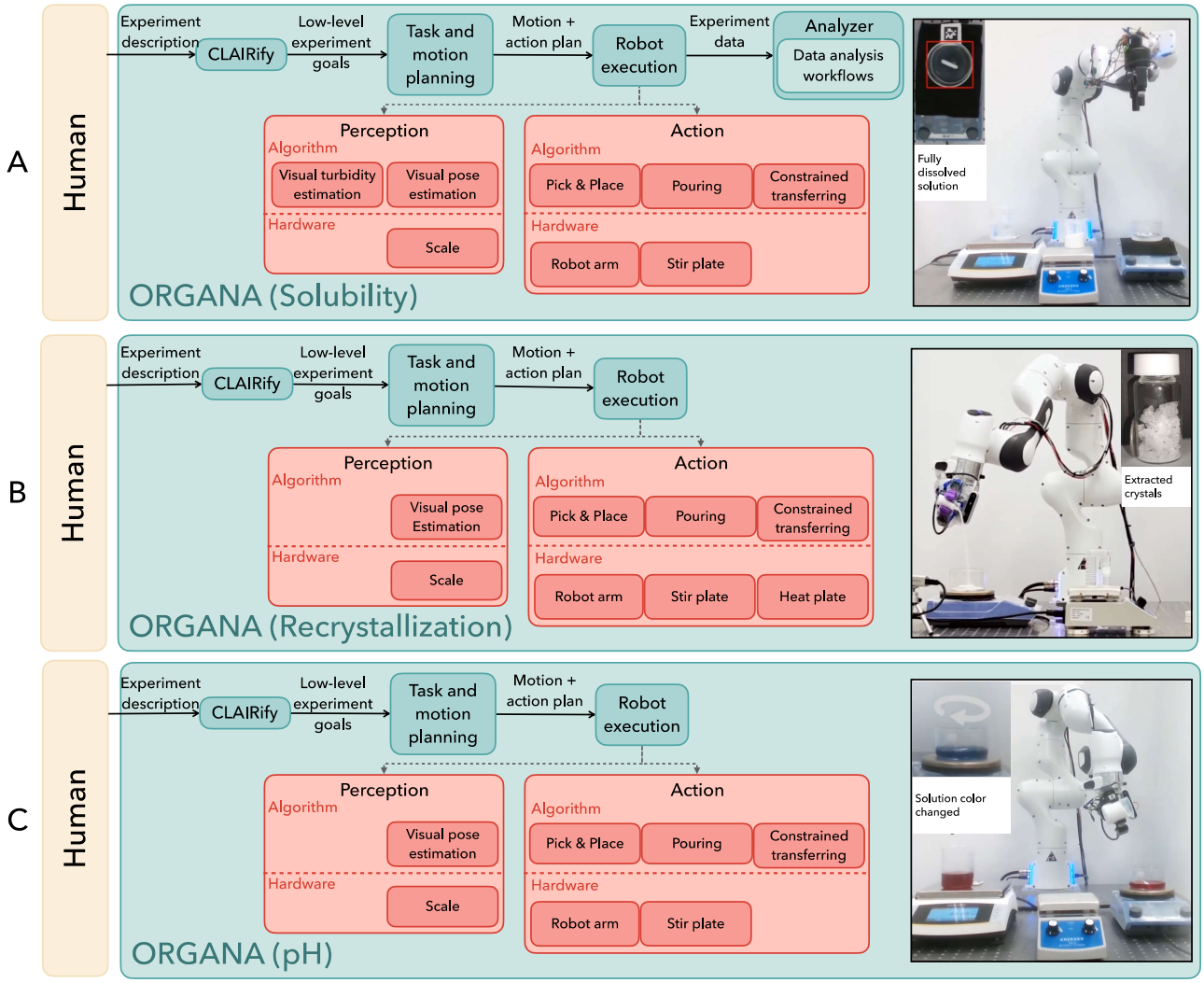


Fig. 4: **Instances of ORGANA conducting various chemistry experiments.** A) solubility, B) recrystallization, C) pH testing. Snapshots depict the robot executing actions in each setup, alongside images of the end results.

value estimation. The system assessed the solubility of three solutes in water —salt (sodium chloride), sugar (sucrose), and alum (aluminum potassium sulfate)— with accuracy values of 7.2%, 11.2%, and 12.3%, respectively, compared to literature results [56]. The main source of error is attributed to the pouring accuracy of the robot [8]. For testing each solution, the system executed a 7-step plan with an average execution time of 25.63 mins.

B. Recrystallization Experiment

Recrystallization is a purification process used to extract pure compounds from impure solids. During this process, impure solids are initially added and dissolved in a solution while being heated, and the process is stopped once saturated. As the solution cools, pure compounds start to crystallize due to the solubility of such compounds decreases, while impurities are maintained in the solution. In our experiment, alum was used to test recrystallization because of its dramatic solubility variations in different water temperatures. This experiment modified the solubility test by pre-heating the solvent. The system performed a 8-step plan with the execution time of 44.80 mins. The result of the robot producing crystals is shown in Figure 4, in the middle.

C. pH Experiment

The pH is a basic property of a solution that is defined by the negative logarithm of the activity of hydrogen ion in an aqueous solution [57]. The anthocyanin pigment in red cabbage can be used as a pH indicator, and the demonstration of its color change is a popular introductory chemistry experiment [58]. We prepared red cabbage solution by boiling red cabbage leaves in hot water. The initial color of the solution was dark purple/red. The color changes to bright pink when an acid is added, and to blue when a base is added. The robot demonstrated this color change by adding food-grade vinegar (acetic acid, an acid) and baking soda (sodium bicarbonate, a base) into the red cabbage solution. We applied the pouring skills of liquid and powder to transfer reagents and presented the color change caused by pH changes as shown in Figure 4.

To perform this experiment, a 6-step plan was executed with a duration of 3.85 minutes.

D. Electrochemistry Experiment

1) Task description: Quinone is an important family of molecules that can be applied to metal-free aqueous flow batteries [59], and their electrochemical properties are actively investigated [60]. The electrochemistry measurements are usually tedious and require human effort for the pretreatment of electrodes which takes several minutes. To demonstrate the applicability of our robotic system in electrochemistry, we measured the redox potential of a quinone solution at different pH levels and drew the Pourbaix diagram. Figure 5 shows our experimental setup. We introduced a portable, low-cost potentiostat [61] and a standard 3-electrode system to conduct the electrochemistry experiments. Notably, we used a glassy-carbon working electrode, which requires mechanical polishing activation, a labor-intensive process in human-centered experiments. Glassy carbon electrode is the most common in electrochemistry studies according to a recent survey [62]. Although mechanical polishing is the most common method for activating glassy carbon electrodes [63], it has not been incorporated into existing automation systems. For example, electrochemical cleaning of the electrode gold surface [64] or disposable microfabricated electrode arrays [10] has been utilized. With our robotic system, we introduced a polishing station together with the robotic arm holding the electrode to automate this process [65].

We prepared quinone solutions in different pH using a flow-based system based on a syringe pump and selection valves, and the redox potential of the solution was measured using a portable potentiostat described in [61]. The solution was a mixture of 2 mM sodium anthraquinone-2-sulfonate from Sigma Aldrich, 0.1 M NaCl, and 0.1 M buffer solution. Six buffer solutions were used for different pH values: acetate buffer (CH_3COONa and CH_3COOH) for pH 4 and 5, phosphate buffer (Na_2HPO_4 and NaH_2PO_4) for pH 6, 7, and 8, and carbonate buffer (Na_2CO_3 and $NaHCO_3$) for pH 9. The pH of the buffer solution was adjusted by mixing two solutions manually. The electrode was mechanically polished for 30 s using a robotic polishing station [65] to ensure activation and washed in deionized water for 30 s to remove residues. In each measurement, three cycles of cyclic voltammetry measurement were conducted in an electrochemical window between -1.5 V and 0.5 V at a scan rate of 100 mV/s. The redox potential was calculated by taking the average of oxidation and reduction peak potentials. The pH of the solution was measured after each electrochemical measurement by automatically transferring the characterized solution to the pH measurement station with the use of a pump.

2) Results: Figure 5 illustrates the setup and workflow employed in conducting electrochemistry experiments. The experimental apparatus encompasses various laboratory equipment, including the polishing station, washing station, pH meter, syringe pump, and robot arm. The experimental environment features three distinct beakers: a large one designated for washing, a small vessel for containing the experiment solution, and another small beaker dedicated to pH measurement. The experimental sequence commences with user interaction with ORGANA to input experiment details. The robot moves toward a view pose, and ORGANA.PERCEPTION detects and estimates the poses of objects within the scene (taking approximately 20 s). The user is prompted to ground beakers and stations, specifying the functionality of the objects present. Upon completing these initial steps, ORGANA.REASONER generates a high-level plan and goal, which is fed into ORGANA.PLANNER to find a parallel executable plan. This process minimizes a cost function associated with total time, thereby maximizing equipment usage. The ORGANA.ROBOTEXECUTION executes the plan through multithreading, as depicted in Figure 5, showcasing snapshots of electrochemistry task progress.

Throughout the experiment, pH values and redox potential values are recorded and provided to the ORGANA.ANALYZER for estimating peak voltage at each pH level. Subsequently, the ORGANA.ANALYZER compares these results with the expected outcomes, initially provided by the user. If inconsistencies arise, ORGANA notifies the user and provides feedback for troubleshooting; otherwise, the experiment proceeds, with the LLM proposing the next buffer solution and the corresponding high-level plan for execution. After testing all buffer solutions, a comprehensive summary report, along with analyzed results, is automatically generated for user review. An example of such a report is provided in Appendix VIII-G.

In total, three complete electrochemistry experiments were performed, testing six buffer solutions for each experiment, ranging from pH 4 to pH 9. Further details regarding the prompts in the human-ORGANA interaction are available in Appendix VIII-B.

a) Quinone Characterization: Quinones typically undergo three different types of reactions depending on the pH of the solution, and they are shown in Figure 6 as the different slopes in the Pourbaix diagram. Two-proton/two-electron reaction is dominant in the region where pH is smaller than pK_{a1}, one-proton/two-electron reaction is dominant between pK_{a1} and pK_{a2}, and zero-proton/two-electron reaction is dominant when pH is larger than pK_{a2} [60]. The redox potential for m proton, n electron redox couple changes $-59\ m/n$ mV/pH unit at 25 °C, according to the Nernst equation [66]. As a consequence, the slope of Pourbaix diagram is predicted to be -59 mV/pH unit in the first region where pH is smaller than pK_{a1}, -30 mV/pH unit in the second region between pK_{a1} and pK_{a2}, and 0 mV/pH unit in the third region where pH is larger than pK_{a2}. The reported dissociation constants of AQS are pK_{a1} = 7.68 and pK_{a2} = 10.92 [66]. Our experimental results agree with these theoretical predictions. In three repeated experiments the estimated values for the slope of the leftmost region were: -61.3, -61.8 and -61.0 mV/pH unit. The estimated values for the dissociation constants pK_{a1} were: 8.12, 7.86 and 8.10. As can be seen in Figure 6, representing the results of a single

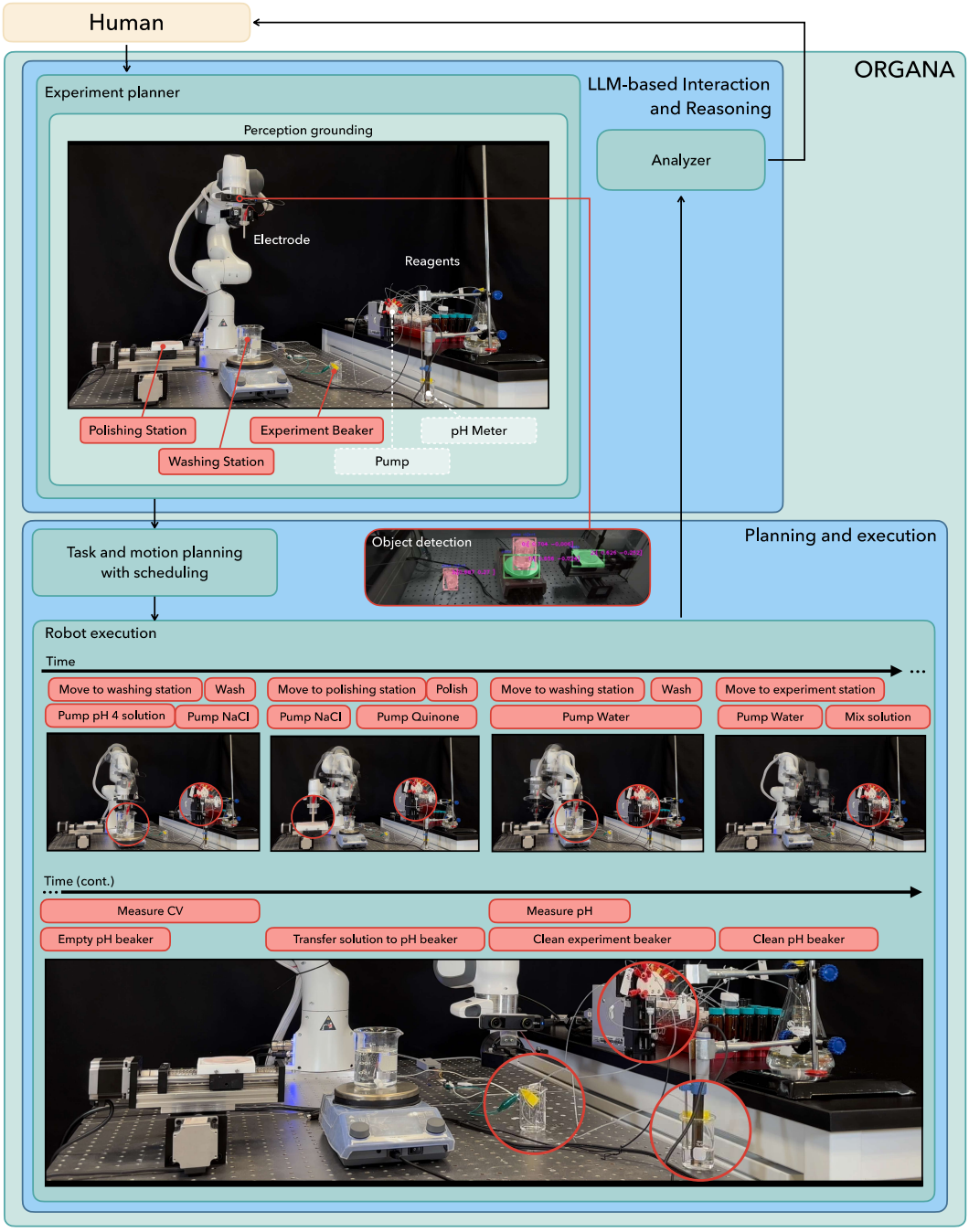


Fig. 5: **The electrochemistry setup and experiment workflow.** Initially, users communicate their intention to ORGANA; objects and their poses are perceived. Subsequently, the user interacts with ORGANA to establish object functionalities. Eventually, ORGANA plans the robot actions for parallel execution. On top, it shows the experiment setup. At the bottom, it displays the results of visual perception and snapshots of the robot and other hardware executing the actions in parallel.

experimental run, even with only 6 measurements we can achieve a low variance estimate for the slope. The variance for $\mathrm{p}K_{\mathrm{a1}}$ value is higher, caused by lack of points for higher pH values. However, even with the given set of pH values, we can produce a lower variance estimate for $\mathrm{p}K_{\mathrm{a1}}$ by utilizing combined data from all three experiments, as can be seen in Figure 8.

b) Sequential and parallel task execution and efficiency: We have performed the electrochemistry experiments both sequentially and with parallel task plans. Figure 7 demonstrates the Gantt chart of the parallel task plan. The

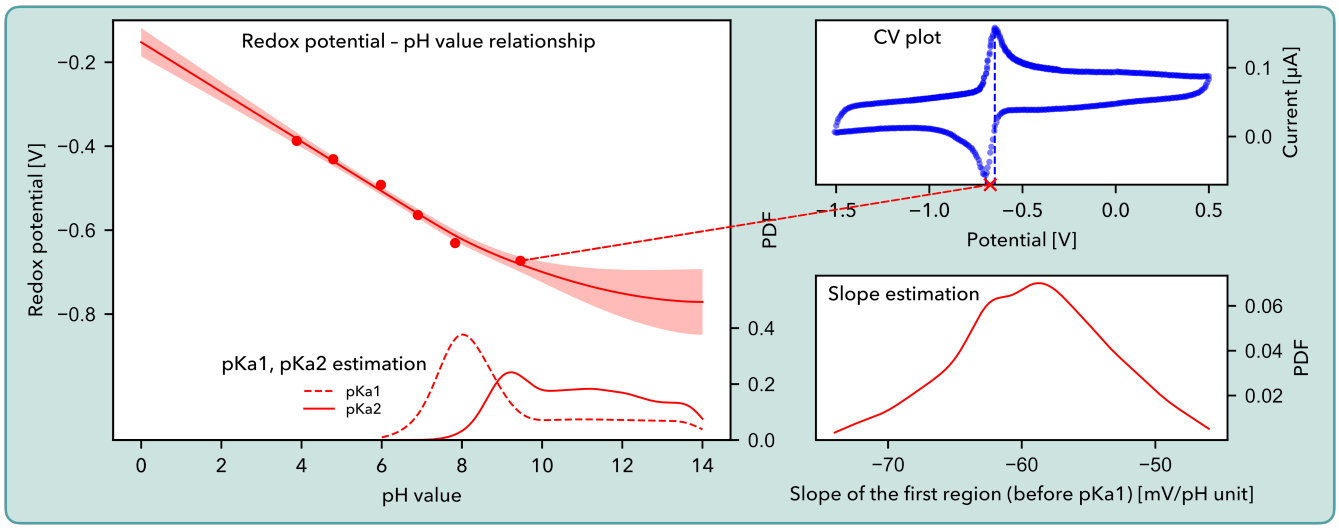


Fig. 6: **The electrochemistry results executed by ORGANA.** On the left, a Pourbaix diagram is shown for a single ORGANA experimental run with estimated distributions for pK_{a1} and pK_{a2}. Maximum likelihood estimation (MLE) for pK_{a1} is 7.86. In the top right, cyclic voltammetry for pH=9 is depicted. In the bottom right, estimated distribution of the slope for the first region is shown, with an MLE of -61.8 mV/pH unit.

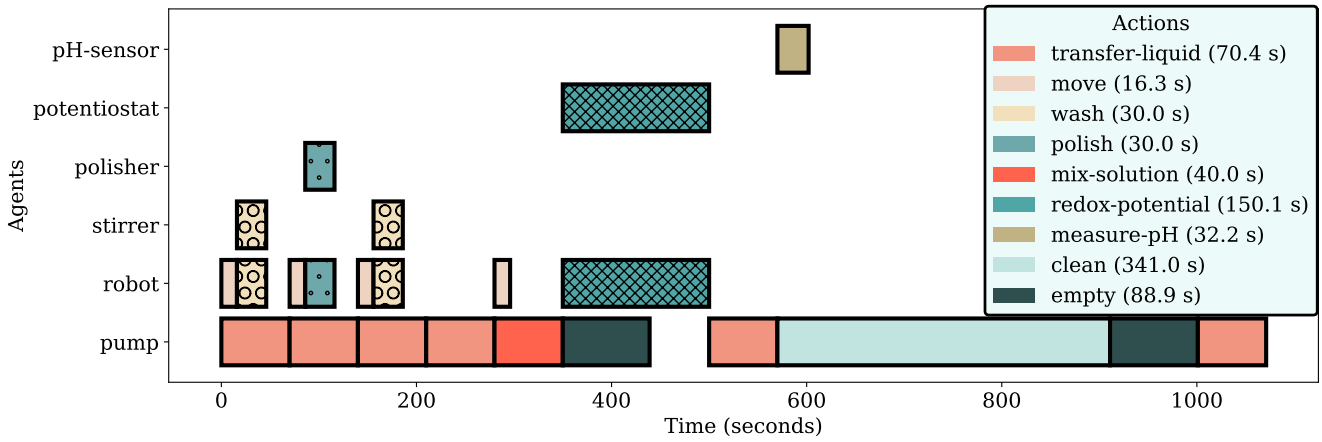


Fig. 7: **Gantt chart of electrochemistry experiment with parallel execution**. The execution times are written in the legend. Note: pump cannot transfer solution to pH beaker at ∼500 sec due to ongoing redox potential measurement. Boxes stacked on top of each other with the same color and pattern mean that the agents are involved in performing the same action jointly.

available agents to perform the actions are indicated on the y-axis of the Gantt chart. When solving the concurrent task and motion planning problem with durative actions, to avoid state space explosion, we assume three possible action durations of 1-3 T for simplification, where T is the action unit time. Actions with a duration of up to 60 seconds are simplified as 1T, 2T (up to 120 seconds), and 3T (more than 180 seconds). In total, the plan is a sequence of 19 actions, some performed by a single agent, while others are performed as a joint action with several agents being involved.

On average, the sequential plan takes 21.67 mins to execute, while the parallel plan takes 17.10 mins to execute, reducing the total time significantly by 21.1%. Finally, the averages and standard deviations of the planning time over 12 trials for the above cases are: solving the sequential planning problem (61.52 ± 0.1 s); and temporal task and motion planning with the time-variant cost function associated with total time (186.3 ± 46.0 s).

E. User Study

To assess the usability of our system in a real-world laboratory setting, we conducted a study with experimental chemists. The details of participants in this study are provided in Appendix VIII-C.

1) Test modes: We asked subjects to perform the following tests:

[T1] Manual experimentation Chemists manually conducted an electrochemistry experiment to generate a Pourbaix plot. Following the procedure outlined in Section IV-D , they prepared a quinone solution, polished an electrode,

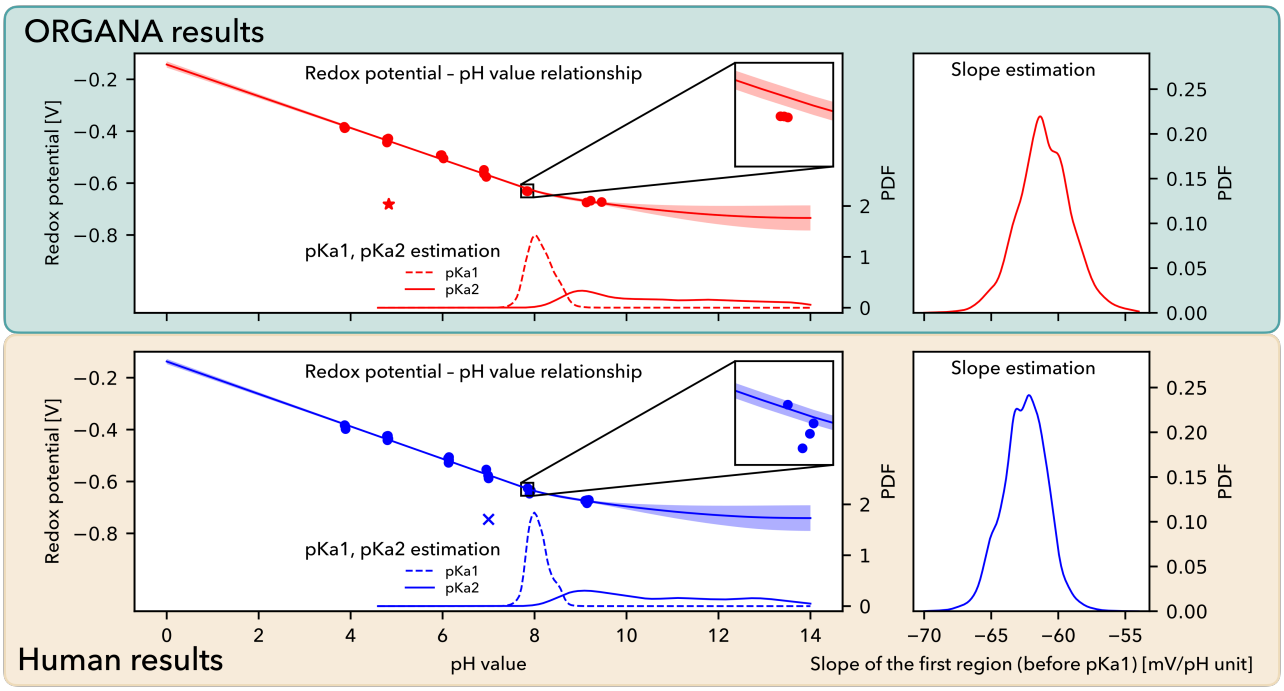


Fig. 8: **Comparison between electrochemistry results conducted by chemists and ORGANA.** Comparison of Pourbaix diagrams and their first region estimated slopes in electrochemistry experiments conducted by ORGANA (top) and chemists (bottom). Results are comparable: ORGANA with pK_{a1}=8.03 and chemists with pK_{a1}=8.02. The estimated slope for ORGANA is -61.3 mV/pH unit and for chemists is -62.7 mV/pH unit. The red star and blue cross symbols highlight two distinct problems with task execution.

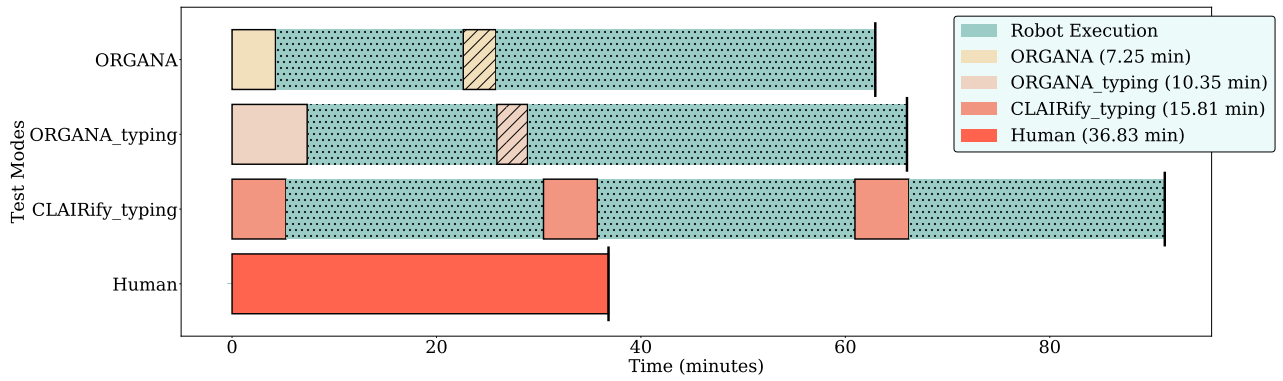


Fig. 9: **Chemist interaction time with ORGANA in various test modes (T1-T4).** The x-axis represents the duration for three iterations of the buffer solution trials in the electrochemistry experiment. This figure shows the timing and frequency of human-ORGANA interactions, with total times specified in the legend. Intermediate user interactions (highlighted with hatched boxes) in ORGANA modes arise from interviewer-introduced troubleshooting, while other modes require user interaction for task execution.

conducted a CV scan, and measured the solution's pH. The experiment was repeated for three buffers with randomly generated pH values.

[T2] ORGANA startup Users interact with ORGANA at the startup phase to provide their intention of the chemistry experiment, i.e., information on experiment procedures, goals, expected observations, and vessel semantics. Chemists were asked to follow a script with the target Pourbaix experiment, first by writing, and then by speaking.

[T3] ORGANA troubleshooting Subjects were asked to interact after obtaining information from the start-up phase. To test user engagement, a scenario was designed where an intentional nonsensical observation triggered ORGANA.REASONER's rationalization component to prompt user feedback, ensuring active involvement despite the system's potential autonomous execution capability.

[T4] CLAIRify Users were instructed to interact with ORGANA as if the language planner module did not exist,

manually detailing each experiment step. This is the equivalent of using CLAIRify [8].

2) Assessment metrics: We assessed both quantitative and qualitative metrics [67], including user interaction time during experiment execution and the variance in Pourbaix plot measurements. Qualitative analysis involved three surveys.

a) NASA Task Load Index (NASA-TLX): It assesses a participant's perceived workload across mental demand, physical demand, temporal demand, effort, performance, and frustration level [68]. Participants self-rated on a scale of 0 (low, good) to 20 (high, bad). The NASA-TLX was administered after both manual experimentation (**T1**) and ORGANA usage (**T2** and **T3**), with the questionnaire details in Appendix VIII-C.

b) System Usability Survey (SUS): It evaluates subjective usability with ten Likert scale questions [69]. Post-interview, scores range from 0 to 10, with higher scores indicating better usability. Users completed the SUS after both manual experimentation (**T1**) and interacting with ORGANA (**T2** and **T3**). The survey questions are detailed in Appendix VIII-C.

c) Custom questionnaire: We created a 24-question custom Likert scale questionnaire to gauge user preferences for various system components and perceptions on lab experiment automation. Questions are framed positively and negatively to alleviate response bias among subjects [70]. The questionnaire is presented in Figure 10.

We performed a one-tailed T-score evaluation at a significance level of $p < 0.05$ for the SUS and workload studies to gauge differences between the manual experiment and ORGANA.

3) User study results:

a) Quantitative analysis: Figure 8 compares Pourbaix plots generated by experimental chemists and ORGANA after the electrochemistry experiment. We compare parameter estimates based on combined data from all human experiments on one hand and all ORGANA experiments on the other. The values are comparable. For $\mathrm{p}K_{\mathrm{a1}}$ ORGANA produces 8.03 and chemists 8.02. For the slope the estimated from ORGANA is -61.3 mV/pH unit, while for the chemists it is -62.7 mV/pH unit.

Figure 9 depicts the time history and frequency of chemists' interactions with ORGANA across different experiment modes (**T1-4**). Notably, ORGANA demonstrates superior performance in terms of human involvement. Manual experimentation (**T1**) averaged over 30 minutes, while ORGANA, during the startup phase, required 6.4 minutes for written and 3.3 minutes for spoken instructions (**T2**). Troubleshooting (**T3**) took chemists an average of 1.3 minutes to provide feedback on errors. Additionally, the CLAIRify-style workflow (**T4**) necessitated an extra 5.3 minutes of user involvement on average after the startup engagement. In 3 experiments with correct results out of 40 total (8 users * 5 experiments without intentional bugs), ORGANA incorrectly alerted the human (false positive). Among 8 experiments with introduced errors, ORGANA failed to detect an issue in only one instance (false negative).

In Figure 8, the star symbol denotes an instance where ORGANA detected and alerted the user to an issue during a chemistry experiment, enabling timely correction. In contrast, the cross in the bottom left figure (**T1**) underscores a scenario where a user omission in a manual user study became apparent only during subsequent data analysis.

b) Qualitative analysis: Figure 10 presents qualitative survey results. In Figure 10(a), NASA-TLX responses for performing manual experiments (**T1**) vs. ORGANA (**T1-2**) reveal reduced demand and effort across all categories. Significantly, ORGANA halved participant frustration and reduced physical demand fourfold. In Figure 10(b), we overlay results from the SUS for the manual experiment and ORGANA for each of the ten questions. We find that ORGANA shows significant improvement compared to manual in the following areas: desire to use the system frequently, reduced complexity of the system, improved consistency, and reduced cumbersomeness. In Figure 10(c), responses to our custom questionnaire indicate unanimous agreement among chemists regarding the potential for automating repetitive lab tasks. Almost all chemists express comfort with robots performing some tasks, with diverse modalities for interaction. Nearly all find ORGANA easy to interact with (only one was neutral) and capable of accurately capturing intended experiments. Preferences for speaking vs. writing to ORGANA vary bimodally. While ORGANA is appreciated for reducing the time that humans need to be involved in the experiment, chemists prefer to be kept informed during experiment execution. The post-experiment summary is unanimously valued by all chemists.

V. DISCUSSION

A. System-Level Discussion

a) Reliability and reproducibility of chemistry results: ORGANA reliably reproduced results from literature for various experiments. In the solubility experiment, ORGANA estimated compound solubility with $10.2 \pm 2.2\%$ mean and standard deviation values. In the electrochemistry experiment (Figure 8), ORGANA produced comparable results to experienced chemists, yielding slope value of $-61.4 \pm 0.5\ mV/pH\ unit$ and $\mathrm{p}K_{\mathrm{a1}}$ value of 8.03 ± 0.17 across three runs of the experiment. These values, as reported in the results section, replicated literature findings. While ORGANA demonstrated proficiency in reproducing literature results in various experiments, challenges persist in tackling complex chemistry tasks requiring advanced perception, manipulation, and planning.

b) Modularity: ORGANA is modular and allows for the independent development and interconnection of modules to obtain and extend the overall functionality. Instances of ORGANA (Figure 4) are applied in four chemistry experiments and integrate variations of NLP, perception, TAMP, robot execution, and data analysis, along with commonly available lab hardware. This modular approach enables rapid customization for new applications, aligning with the concept of material on demand for lab automation. Utilizing a general-purpose robot equipped with a multi-modal large

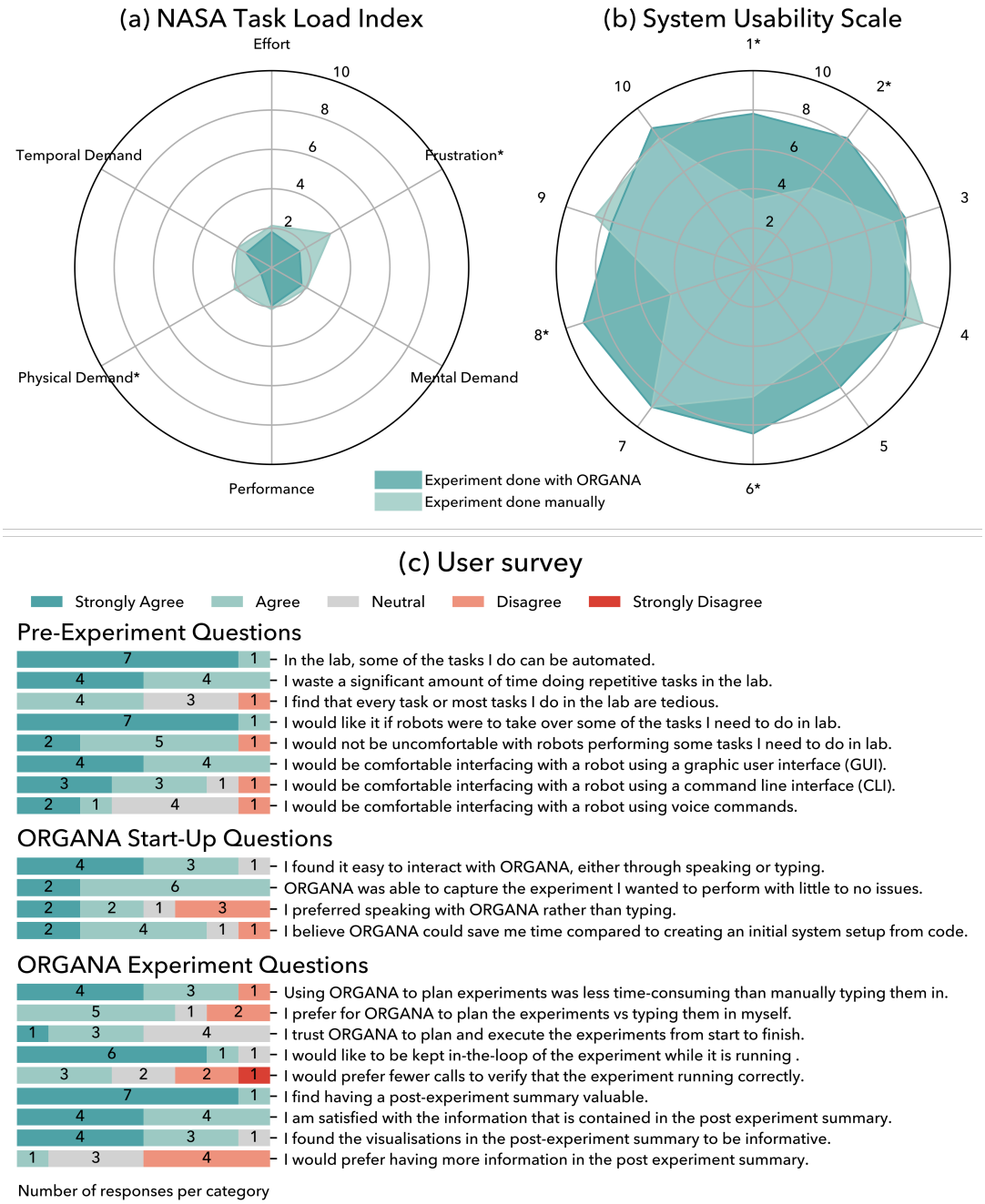


Fig. 10: **Human subjective measure figure.** (a) NASA Task Load Index (the lower, the better). (b) System Usability Scale (the higher, the better). (c) User study questions. Asterisks in (a) and (b) indicate a significant improvement of ORGANA over the manual experiment for specified metrics.

perception and manipulation model can potentially enhance flexibility, reducing the need for specialized algorithms in novel applications [71]. In electrochemistry experiments, we leverage LLMs and a transformer-based visual perception architecture for high-level task planning and object perception. In contrast to other lab automation projects where robots operate in structured environments with fixed object poses, ORGANA relaxes this assumption. It perceives and acts in a semi-structured environment where objects and their poses can vary.

c) Evaluation of autonomy and robustness: ORGANA successfully completed four different long-horizon chemistry experiments, including:

- solubility: 7 steps plan with 25.63 mins execution time performed for 2 times;
- recrystallization: 8 steps plan with 44.80 mins execution time performed for 1 time;
- pH measurement: 6 steps plan with 3.85 mins execution

time performed for 1 time;
- electrochemistry: 114(6×19) steps plan with 130.00 mins execution time performed 2 times.

The human-in-the-loop feature of ORGANA makes it more robust to failures and reliable through timely interaction with users. In Figure 8, the star indicates a scenario where ORGANA detected and informed the user about an issue during a chemistry experiment, enabling timely correction.

d) Efficiency and support for high-throughput experimentation: ORGANA increases efficiency and maximizes the usage of resources in chemistry experiments by parallelizing the execution of tasks using the available resources, i.e., pump, robot arm, stirrer, polisher, potentiostat, and pH-sensor. This is achieved by solving the TAMP and scheduling problems together, which can lead to accelerated material discovery, a crucial component that involves supporting high-throughput experimentation and screening. In this work, ORGANA.PLANNER minimizes a cost function tied to total time; however, it is also possible to consider additional costs associated with the quality of task execution. As demonstrated in Section IV-D , ORGANA.PLANNER results in a notable enhancement of 274 seconds (21.1%) in the overall electrochemistry time compared to sequential task execution. Solving TAMP and scheduling problems together incur an overhead compared to sequential TAMP during the planning phase. Employing learning-based techniques or LLMs is a potential approach for reducing planning time [29, 72].

e) Safety: To address safety in lab automation, aside from relying on having a human in the loop for disambiguation, ORGANA relies on constrained motion planning, consistency checks, and feedback integration.

Constrained motion planning was applied and proved to be critical in solubility and recrystallization experiments in order to prevent spills while moving vials between locations. ORGANA, equipped with LLMs for reasoning, has the capability to detect unexpected events and address them through user interaction. This feature holds promise for enhancing safety when safety rationales are provided to it by the human. Moreover, ORGANA's report generation feature can play a crucial role in documenting and keeping scientists informed in case of safety violations. For example, in the report, a section could be dedicated to safety-related notes, according to safety rationales and metrics. Finally, comprehensive safety should extend to both the physical and psychological well-being of humans, taking into account chemical, electrical, and mechanical hazards during synthesis and reactions. This necessitates preemptive and post-event safety measures in perception, planning, and execution, facilitating timely adjustments.

B. Discussion of Interactions Between Humans and ORGANA

The user study indicates participants found lab automation, specifically ORGANA, useful. They expressed comfort with various communication modalities such as command line interface (CLI), GUI, or natural language.

The study demonstrated a significant reduction in user physical load and frustration during chemistry experimentation with ORGANA. Participants consistently rated ORGANA as significantly useful (SUS, questions 1, 2) and expressed satisfaction (SUS, questions 8, 9) with its performance in chemistry experiments. The findings suggest that enhancing ORGANA's ease of use and learning could further streamline system usability.

While users did not perceive a significant increase in efficiency based on workload and SUS studies, Figure 9 indicates the quantitative importance of ORGANA in reducing human temporal workload. This discrepancy might stem from users performing only a part of the full experiment. Nevertheless, this result highlights the potential for system improvement.

Users agree on the necessity of keeping humans in the loop of autonomy. In fact, half the users expressed uncertainty in trusting a robot to complete the experiments autonomously from start to finish. This indicates the need for careful consideration when integrating robots into human workflows. Comprehensive report generation was identified as one potential method of increasing trust, as well as pinging humans while the experiment was being executed at moments of uncertainty.

C. Limitations

ORGANA currently relies primarily on independent sensor modalities for perception. There is potential in exploring multimodal perception to monitor the progress of chemistry tasks and enhance decision-making. An example can be found in [73] to monitor chemistry task progress, where several process parameters (such as temperature and stir rate) and visual cues (such as volume, color, turbidity) were combined to control a chemistry process inside a reactor. Another instance of multimodal perception involves using haptic feedback and vision for object manipulation [74, 75]. Finally, although we addressed the challenge of transparent object detection and pose estimation for our setup in electrochemistry experiments, there is much to be done to develop a robust model-free transparent object perception, considering their textureless and reflective surface [43].

A limitation of ORGANA.PLANNER lies in the complexity of defining a PDDL domain for planning robot actions, potentially making it challenging for chemists who are not experts in planning and robotics to modify it. Additionally, it lacks support for online replanning, limiting its adaptability to uncertainties in task execution. We are working on LLMs for task planning and replanning, which holds promise for solving multi-stage long-horizon tasks, but validating the proposed LLM plans and ensuring generalization remains a challenge [72, 76–79]. In past work, we have started to address this by learning planning heuristics from experience [29, 80].

VI. CONCLUSION

This paper introduces ORGANA, a user-friendly robotic assistant for lab automation system designed for chemistry

experiments. ORGANA effectively executes long-horizon experiments, interacting with users in natural language to identify objectives and providing comprehensive reports. It accommodates various chemistry experiments in a semi-structured environment, enabling parallel planning and execution through solving scheduling and task and motion planning together. ORGANA's capabilities were validated in an electrochemistry experiment on quinone characterization, a compound in rechargeable flow batteries. The user study indicates chemists find ORGANA and its report generation useful for lab automation.

REFERENCES

1. M. Christensen, L. P. Yunker, P. Shiri, T. Zepel, P. L. Prieto, S. Grunert, F. Bork, and J. E. Hein, "Automation isn't automatic," *Chemical Science*, vol. 12, no. 47, pp. 15 473–15 490, 2021.
2. L. M. Roch, F. Häse, C. Kreisbeck, T. Tamayo-Mendoza, L. P. Yunker, J. E. Hein, and A. Aspuru-Guzik, "Chemos: orchestrating autonomous experimentation," *Science Robotics*, vol. 3, no. 19, 2018.
3. S. H. M. Mehr, M. Craven, A. I. Leonov, G. Keenan, and L. Cronin, "A universal system for digitization and automatic execution of the chemical synthesis literature," *Science*, vol. 370, no. 6512, pp. 101–108, 2020.
4. B. Burger, P. M. Maffettone, V. V. Gusev, C. M. Aitchison, Y. Bai, X. Wang, X. Li, B. M. Alston, B. Li, R. Clowes *et al.*, "A mobile robotic chemist," *Nature*, vol. 583, no. 7815, pp. 237–241, 2020.
5. R. Vescovi, T. Ginsburg, K. Hippe, D. Y. Ozgulbas, C. Stone, A. Stroka, R. Butler, B. J. Blaiszik, T. Brettin, K. Chard *et al.*, "Towards a modular architecture for science factories," *Digital Discovery*, 2023.
6. B. P. MacLeod, F. G. Parlane, A. K. Brown, J. E. Hein, and C. P. Berlinguette, "Flexible automation accelerates materials discovery," *Nature Materials*, vol. 21, no. 7, pp. 722–726, 2022.
7. C. Steinruecken, E. Smith, D. Janz, J. Lloyd, and Z. Ghahramani, "The automatic statistician," *Automated machine learning: Methods, systems, challenges*, pp. 161–173, 2019.
8. N. Yoshikawa, M. Skreta, K. Darvish, S. Arellano-Rubach, Z. Ji, L. Bjørn Kristensen, A. Z. Li, Y. Zhao, H. Xu, A. Kuramshin *et al.*, "Large language models for chemistry robotics," *Autonomous Robots*, pp. 1–30, 2023.
9. S. Steiner, J. Wolf, S. Glatzel, A. Andreou, J. M. Granda, G. Keenan, T. Hinkley, G. Aragon-Camarasa, P. J. Kitson, D. Angelone *et al.*, "Organic synthesis in a modular robotic system driven by a chemical programming language," *Science*, vol. 363, no. 6423, p. eaav2211, 2019.
10. I. Oh, M. A. Pence, N. G. Lukhanin, O. Rodríguez, C. M. Schroeder, and J. Rodríguez-López, "The electrolab: An open-source, modular platform for automated characterization of redox-active electrolytes," *Device*, vol. 1, no. 5, p. 100103, 2023.
11. D. Knobbe, H. Zwirnmann, M. Eckhoff, and S. Haddadin, "Core processes in intelligent robotic lab assistants: Flexible liquid handling," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2335–2342.
12. H. Fakhruldeen, G. Pizzuto, J. Glowacki, and A. I. Cooper, "Archemist: Autonomous robotic chemistry system architecture," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6013–6019.
13. S. Eppel, H. Xu, M. Bismuth, and A. Aspuru-Guzik, "Computer vision for recognition of materials and vessels in chemistry lab settings and the vector-labpics data set," *ACS Central Science*, vol. 6, no. 10, pp. 1743–1752, 2020. [Online]. Available: https://doi.org/10.1021/acscentsci.0c00460
14. R. El-khawaldeh, M. A. Guy, F. Bork, N. Taherimakhsousi, K. N. Jones, J. Hawkins, L. Han, R. P. Pritchard, B. Cole, S. Monfette, and J. E. Hein, "Keeping an "eye" on the experiment: computer vision for real-time monitoring and control," *Chem. Sci.*, pp. –, 2023. [Online]. Available: http://dx.doi.org/10.1039/D3SC05491H
15. T. Zepel, V. Lai, L. P. E. Yunker, and J. E. Hein, "Automated liquid-level monitoring and control using computer vision," *ChemRxiv*, 2020.
16. H. Xu, Y. R. Wang, S. Eppel, A. Aspuru-Guzik, F. Shkurti, and A. Garg, "Seeing glass: Joint point cloud and depth completion for transparent objects," *arXiv preprint arXiv:2110.00087*, 2021.
17. Y. R. Wang, Y. Zhao, H. Xu, S. Eppel, A. Aspuru-Guzik, F. Shkurti, and A. Garg, "Mvtrans: Multi-view perception of transparent objects," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3771–3778.
18. N. J. Szymanski, B. Rendy, Y. Fei, R. E. Kumar, T. He, D. Milsted, M. J. McDermott, M. Gallant, E. D. Cubuk, A. Merchant *et al.*, "An autonomous laboratory for the accelerated synthesis of novel materials," *Nature*, pp. 1–6, 2023.
19. R. Duke, S. Mahmoudi, A. P. Kaur, V. Bhat, I. Dingle, N. C. Stumme, S. K. Shaw, D. Eaton, A. Vego, and C. Risko, "Expflow: a graphical user interface for automated reproducible electrochemistry," *Digital Discovery*, 2024.
20. A. Aspuru-Guzik, R. Lindh, and M. Reiher, "The matter simulation (r) evolution," *ACS central science*, vol. 4, no. 2, pp. 144–152, 2018.
21. A. C. Vaucher, P. Schwaller, J. Geluykens, V. H. Nair, A. Iuliano, and T. Laino, "Inferring experimental procedures from text-based representations of chemical reactions," *Nature communications*, vol. 12, no. 1, p. 2573, 2021.
22. Z. Ren, Z. Zhang, Y. Tian, and J. Li, "Crest – copilot for real-world experimental scientist," *ChemRxiv*, 2023.
23. A. M. Bran, S. Cox, A. D. White, and P. Schwaller, "Chemcrow: Augmenting large-language models with

chemistry tools," *arXiv preprint arXiv:2304.05376*, 2023.
24. D. A. Boiko, R. MacKnight, B. Kline, and G. Gomes, "Autonomous chemical research with large language models," *Nature*, 2023.
25. C. R. Garrett, T. Lozano-Pérez, and L. P. Kaelbling, "PDDLStream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning," in *Proceedings of the 30th Int. Conf. on Automated Planning and Scheduling (ICAPS)*. AAAI Press, 2020, pp. 440–448.
26. D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins, "Pddl-the planning domain definition language," Technical Report CVC TR98003/DCS TR1165. New Haven, CT: Yale Center for Computational Vision and Control, Tech. Rep. 123, 1998. [Online]. Available: http://www.example.com/advancements_report
27. M. Toussaint, "Logic-geometric programming: An optimization-based approach to combined task and motion planning." in *IJCAI*, 2015, pp. 1930–1936.
28. M. A. Toussaint, K. R. Allen, K. A. Smith, and J. B. Tenenbaum, "Differentiable physics and stable modes for tool-use and manipulation planning," *Robotics: Science and Systems Foundation*, 2018.
29. M. Khodeir, B. Agro, and F. Shkurti, "Learning to search in task and motion planning with streams," *IEEE Robotics and Automation Letters*, 2023.
30. B. Kim, L. Shimanuki, L. P. Kaelbling, and T. Lozano-Pérez, "Representation, learning, and planning algorithms for geometric task and motion planning," *The International Journal of Robotics Research*, vol. 41, no. 2, pp. 210–231, 2022.
31. N. Kumar, W. McClinton, R. Chitnis, T. Silver, T. Lozano-Pérez, and L. P. Kaelbling, "Learning efficient abstract planning models that choose what to predict," in *7th Annual Conference on Robot Learning*, 2023.
32. F. Häse, L. M. Roch, and A. Aspuru-Guzik, "Next-generation experimentation with self-driving laboratories," *Trends in Chemistry*, vol. 1, no. 3, pp. 282–291, 2019.
33. C. D. Hubbs, C. Li, N. V. Sahinidis, I. E. Grossmann, and J. M. Wassick, "A deep reinforcement learning approach for chemical production scheduling," *Computers & Chemical Engineering*, vol. 141, p. 106982, 2020.
34. D. Long, J. Dolejsi, and M. Stolba, "Scheduling problems in pddl," in *Workshop on Knowledge Engineering for Planning and Scheduling*, 2023.
35. M. Fox and D. Long, "Pddl2. 1: An extension to pddl for expressing temporal planning domains," *Journal of artificial intelligence research*, vol. 20, pp. 61–124, 2003.
36. S. Edelkamp, M. Lahijanian, D. Magazzeni, and E. Plaku, "Integrating temporal reasoning and sampling-based motion planning for multigoal problems with dynamics and time windows," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3473–3480, 2018.
37. J. Chen, B. C. Williams, and C. Fan, "Optimal mixed discrete-continuous planning for linear hybrid systems," in *Proceedings of the 24th International Conference on Hybrid Systems: Computation and Control*, ser. HSCC '21. New York, NY, USA: Association for Computing Machinery, 2021. [Online]. Available: https://doi.org/10.1145/3447928.3456654
38. S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. R. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models," in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=WE_vluYUL-X
39. M. Skreta, N. Yoshikawa, S. Arellano-Rubach, Z. Ji, L. B. Kristensen, K. Darvish, A. Aspuru-Guzik, F. Shkurti, and A. Garg, "Errors are useful prompts: Instruction guided task programming with verifier-assisted iterative prompting," *arXiv preprint arXiv:2303.14100*, 2023.
40. A. Majumdar, F. Xia, B. Ichter, D. Batra, and L. Guibas, "Findthis: Language-driven object disambiguation in indoor environments," in *Conference on Robot Learning*. PMLR, 2023, pp. 1335–1347.
41. M. Helmert, "The fast downward planning system," *Journal of Artificial Intelligence Research*, vol. 26, pp. 191–246, 2006.
42. J. Illingworth and J. Kittler, "A survey of the hough transform," *Computer vision, graphics, and image processing*, vol. 44, no. 1, pp. 87–116, 1988.
43. J. Jiang, G. Cao, J. Deng, T.-T. Do, and S. Luo, "Robotic perception of transparent objects: A review," *IEEE Transactions on Artificial Intelligence*, 2023.
44. S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu, and L. Zhang, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," 2023.
45. H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "Dino: Detr with improved denoising anchor boxes for end-to-end object detection," *arXiv preprint arXiv:2203.03605*, 2022.
46. Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE*, vol. 111, no. 3, pp. 257–276, 2023.
47. A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.
48. A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
49. Stereolabs, "ZED 2 - AI Stereo Camera," https://www.stereolabs.com/products/zed-2, accessed: 2023-12-22.
50. Q.-Y. Zhou, J. Park, and V. Koltun, "Open3D: A modern library for 3D data processing," *arXiv preprint arXiv:1801.09847*, 2018.

51. C. Labrín and F. Urdinez, "Principal component analysis," in *R for Political Data Science*. Chapman and Hall/CRC, 2020, pp. 375–393.
52. P. Beeson and B. Ames, "Trac-ik: An open-source library for improved solving of generic inverse kinematics," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 928–935.
53. Z. Kingston, M. Moll, and L. E. Kavraki, "Exploring implicit spaces for constrained sampling-based planning," *Int. J. Robot. Res.*, vol. 38, no. 10-11, pp. 1151–1178, 2019.
54. S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.
55. "redox potential," *International Union of Pure and Applied Chemistry (IUPAC)*, 2019. [Online]. Available: https://doi.org/10.1351/goldbook.RT06783
56. N. A. O. o. J. NAOJ, *Handbook of Scientific Tables*. WORLD SCIENTIFIC, 2022.
57. "ph," *International Union of Pure and Applied Chemistry (IUPAC)*, 2019. [Online]. Available: https://doi.org/10.1351/goldbook.P04524
58. J. J. Fortman and K. M. Stubbs, "Demonstrations with red cabbage indicator," *Journal of chemical education*, vol. 69, no. 1, p. 66, 1992.
59. B. Huskinson, M. P. Marshak, C. Suh, S. Er, M. R. Gerhardt, C. J. Galvin, X. Chen, A. Aspuru-Guzik, R. G. Gordon, and M. J. Aziz, "A metal-free organic–inorganic aqueous flow battery," *Nature*, vol. 505, no. 7482, pp. 195–198, 2014.
60. A. Khetan, "High-throughput virtual screening of quinones for aqueous redox flow batteries: Status and perspectives," *Batteries*, vol. 9, no. 1, p. 24, 2022.
61. S. Pablo-García *et al.*, 2024, (Manuscript in preparation).
62. D. M. Heard and A. J. Lennox, "Electrode materials in modern organic electrochemistry," *Angewandte Chemie International Edition*, vol. 59, no. 43, pp. 18 866–18 884, 2020.
63. G. M. Swain, "Solid electrode materials: pretreatment and activation," in *Handbook of electrochemistry*. Elsevier, 2007, pp. 111–153.
64. K. Laws, M. Tze-Kiat Ng, A. Sharma, Y. Jiang, A. J. S. Hammer, and L. Cronin, "An autonomous electrochemical discovery robot that utilises probabilistic algorithms: Probing the redox behaviour of inorganic materials," *ChemElectroChem*, p. e202300532, 2023.
65. N. Yoshikawa *et al.*, 2024, (Manuscript in preparation).
66. M. Quan, D. Sanchez, M. F. Wasylkiw, and D. K. Smith, "Voltammetry of quinones in unbuffered aqueous solution: reassessing the roles of proton transfer and hydrogen bonding in the aqueous electrochemistry of quinones," *Journal of the American Chemical Society*, vol. 129, no. 42, pp. 12 847–12 856, 2007.
67. K. Darvish, L. Penco, J. Ramos, R. Cisneros, J. Pratt, E. Yoshida, S. Ivaldi, and D. Pucci, "Teleoperation of humanoid robots: A survey," *IEEE Transactions on Robotics*, 2023.
68. S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 50. Sage publications Sage CA: Los Angeles, CA, 2006, pp. 904–908.
69. J. Brooke, "Sus: A quick and dirty usability scale," *Usability Eval. Ind.*, vol. 189, 11 1995.
70. A. Furnham, "Response bias, social desirability and dissimulation," *Personality and individual differences*, vol. 7, no. 3, pp. 385–400, 1986.
71. A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.
72. B. Liu, Y. Jiang, X. Zhang, Q. Liu, S. Zhang, J. Biswas, and P. Stone, "Llm+ p: Empowering large language models with optimal planning proficiency," *arXiv preprint arXiv:2304.11477*, 2023.
73. R. El-khawaldeh, M. A. Guy, F. Bork, N. Taherimakhsousi, K. N. Jones, J. Hawkins, L. Han, R. P. Pritchard, B. Cole, S. Monfette *et al.*, "Keeping an "eye" on the experiment: computer vision for real-time monitoring and control," *Chemical Science*, 2023.
74. P. K. Murali, B. Porr, and M. Kaboli, "Touch if it's transparent! actor: Active tactile-based category-level transparent object reconstruction," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 10 792–10 799.
75. M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg, "Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8943–8950.
76. M. Skreta, Z. Zhou, J. L. Yuan, K. Darvish, A. Aspuru-Guzik, and A. Garg, "Replan: Robotic replanning with perception and language models," 2024.
77. I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, "Progprompt: Program generation for situated robot task planning using large language models," *Autonomous Robots*, pp. 1–14, 2023.
78. J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," *arXiv preprint arXiv:2209.07753*, 2022.
79. D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.
80. M. Khodeir, A. Sonwane, R. Hari, and F. Shkurti, "Policy-guided lazy search with feedback for task and motion planning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3743–3749.
81. M. Walker, G. Pizzuto, H. Fakhruldeen, and A. I.

Cooper, "Go with the flow: deep learning methods for autonomous viscosity estimations," *Digital Discovery*, vol. 2, pp. 1540–1547, 2023. [Online]. Available: http://dx.doi.org/10.1039/D3DD00109A
82. E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *2011 IEEE international conference on robotics and automation*. IEEE, 2011, pp. 3400–3407.
83. G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.
84. C. Hwang, W. Cui, Y. Xiong, Z. Yang, Z. Liu, H. Hu, Z. Wang, R. Salas, J. Jose, P. Ram, J. Chau, P. Cheng, F. Yang, M. Yang, and Y. Xiong, "Tutel: Adaptive mixture-of-experts at scale," *arXiv preprint arXiv:2206.03382*, 2022.
85. A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=YicbFdNTTy
86. Stereolabs, "ZED Mini Camera and SDK Overview," 2019.
87. N. Elgrishi, K. J. Rountree, B. D. McCarthy, E. S. Rountree, T. T. Eisenhart, and J. L. Dempsey, "A practical beginner's guide to cyclic voltammetry," *Journal of chemical education*, vol. 95, no. 2, pp. 197–206, 2018.
88. H. E. Grecco, M. C. Dartiailh, G. Thalhammer-Thurner, T. Bronger, and F. Bauer, "Pyvisa: the python instrumentation package," *Journal of Open Source Software*, vol. 8, no. 84, p. 5304, 2023.

VII. Acknowledgments

We thank members of the Matter Lab for participating in the user study. We would like to also thank Jinbang Huang, Lasse Bjørn Kristensen, and Jason Hattrick-Simpers for their insightful discussions, and the Acceleration Consortium for their generous support.

a) Funding: The paper was supported by the University of Toronto.

b) Authors contributions: K.D. led the technical development and validation of the system and contributed to task and motion planning with the scheduling. M.S. contributed to developing interaction and reasoning and the user study. Y.Z. contributed to system integration and developing perception, and task and motion planning with the scheduling. N.Y. contributed to developing the chemistry experiment protocol and system development. S.S. contributed to developing perception and system integration. M.B. contributed to developing parameter estimation. H.H. and Y.C. developed the chemistry experiment protocol. H.X. contributed to developing perception. A.A.G. supervised chemistry experiment protocols and speech interaction. A.G. supervised large language model-based planning and reasoning. F.S. supervised task and motion planning with scheduling, user study, and automated report generation. K.D., M.S., Y.Z., N.Y., S.S., and M.B. wrote the initial draft. All the authors proofread the manuscript.

VIII. Appendices

A. Perception Analysis

a) Perception for Lab Automation Background: Effective scene perception is paramount in the lab automation context, with a particular emphasis on two facets of perceptual skills: chemistry-level perception for synthesis monitoring and analysis and object-level perception for the manipulation of lab equipment. Addressing chemistry-level perception, early efforts, exemplified by LabPics [13], introduced an innovative image dataset and employed convolutional neural networks (CNNs) to discern material phases and delineate phase boundaries. Recent advancements by HeinSight systems [14, 15] have furthered this field by presenting a more generalized vision system that not only classifies material phases but also quantifies physical properties such as volume, color, and turbidity, thereby facilitating comprehensive monitoring and control of experimental processes. Moreover, in the work of [81], liquid viscosity estimation was achieved through the utilization of robot manipulators for collecting fluid motion videos, followed by 3D CNN analysis.

Turning to object-level perception, the transparency of many chemistry lab tools poses a unique challenge. Notably, efforts by [16, 17] focused on detecting transparent vessels in occluded scenes by leveraging 3D reconstruction and multiview perception. This approach enabled the estimation of depth, segments, and object poses, enhancing the efficacy of downstream manipulation tasks. These models focus on detecting common transparent and opaque objects; however, they do not generalize well to unfamiliar and novel objects. We adopted GroundDINO with SAM, a more versatile approach that is also capable of accepting text prompts.

b) Evaluation of object detection and pose estimation in ORGANA*:* The perception pipeline is evaluated in two aspects: object detection and object position estimation. The Average Precision was used to measure the detection quality for each object class. Mean Absolute Error was used to evaluate the position estimate of the objects. Since we assumed that objects maintain an upright orientation and object orientation is not utilized in our chemistry experiments, the quality of the orientation prediction was not evaluated. A real dataset related to electrochemistry setup was constructed for conducting these evaluations.

c) Data Collection: We use the ZED Mini camera with known intrinsic parameters to collect images with different scene setups. Specifically, chemistry equipment, including beakers, flasks, and a polishing pad are placed on the table. We randomize lighting conditions, table backgrounds, colored liquids in transparent vessels, and object locations. We automate the data collection by having the robot with an eye-in-hand camera capture RGBD images from different angles as shown in Figure 11. In total, the dataset consists of 135 RGBD images captured across 17 different scenes. Each scene contains 4 transparent objects and 1 polishing plate.

To obtain the ground truth object pose in the world frame, AprilTags [82] are randomly placed on the table. An image is then captured by the robot camera, and tag 6D poses are estimated using OpenCV functions [83]. We convert these poses into world coordinates based on known camera poses and subsequently replace the tags with associated objects on the table. For obtaining the ground truth 2D bounding boxes for each object, we create 3D mesh models based on measured dimensions. Point clouds of these mesh models are projected to 2D image space using the corresponding tag pose, camera intrinsic and extrinsic matrices. The 2D bounding box for each set of projected points is generated and treated as the ground truth.

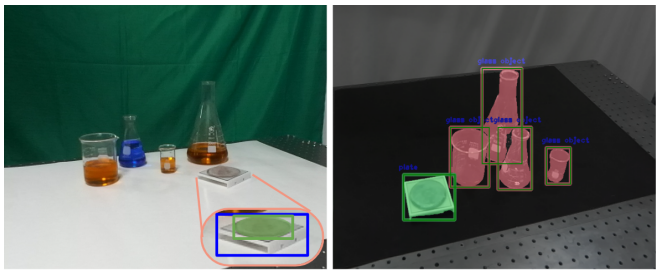


Fig. 11: **Evaluation of ORGANA.PERCEPTION on two sample images.** Left: A sample image from the dataset showing how the objects are arranged. For the polishing pad, the blue bounding box represents the ground truth whereas the green box represents the predicted output from the perception pipeline. Right: An example of detection and segmentation of partially occluded vessels and the polishing plate.

d) Implementation: As stated in the Perception Section of the paper, the perception pipeline is built based on GroundDINO and SAM models with a custom object pose estimation mechanism shown in Figure 12. For evaluation, we directly use the checkpoint of the GroundDINO model [44] with the Swin-T backbone [84] and SAM [47] with the ViT-B backbone [85]. To detect beakers and the polishing station in the scene, we prompt the GroundDINO with "glass object" and "plate", as demonstrated to be robust based on the results section below.

e) Results: For object detection, Table I summarizes the Average Precision (AP) for each class supported by the perception pipeline. We observed that the precision of glass objects is consistently high across different Intersection over Union (IoU) thresholds. In contrast, the pipeline achieves a high AP for detecting plates with low IoU thresholds, but the detection performance dramatically decreases as IoU increases. This occurs because the GroundningDINO model often recognizes the brown pad as the plate rather than the entire polishing pad fixture, resulting in a lower IoU, as demonstrated in Figure 11. Additionally, we found that the pipeline can handle partial occlusion, as shown in Figure 11. This capability enables the pipeline to withstand more realistic and complex conditions outside of preset object configurations.

TABLE I: Average Precision achieved by the perception pipeline on each of the supported classes for different IoU thresholds

IoU Threshold	AP of glass	AP of plate
0.25	94.9	81.4
0.50	93.9	38.6
0.75	90.7	37.1

In terms of object position estimation, the pipeline achieved a Mean Absolute Error (MAE) of 3.5 cm, averaged over all objects in the dataset. Specifically, the small beaker has the lowest MAE at 2.4 cm, while the large flask has the highest MAE at 5.1 cm. The accuracy of the object position is also highly dependent on the depth map from the ZED camera, as discussed in the Perception Section of the paper. Although the ZED Mini camera is reported to have a depth error of 1.5% within its range of 10 cm to 3 m [86], the depth accuracy of transparent objects degrades given that background depth values are reported instead. To address this issue, we applied the radius outlier removal method [50] directly on point clouds as demonstrated in Figure 13, improving the overall MAE from 4.5 cm to 3.5 cm.

Based on these results, it is clear that the perception module is reliable and robust under different conditions in our lab setting. Future work will involve developing the module to support a diverse set of lab equipment while achieving high precision in pose estimation with image input.

B. ORGANA.REASONER *Implementation and Prompts*

Text and speech interface: Users can interact with ORGANA.REASONER using a text interface or speech. The speech interface detects when humans are speaking using the SpeechRecognition* library in Python. Speech is processed to text using OpenAI's Whisper model†. OpenAI's GPT-4‡ is used to reason over the user input and generate follow-up questions, which are processed to speech using ElevenLabs voice generators§.

ORGANA.REASONER *prompts:* The chemist interacts with ORGANA in three key stages: [1] Experiment Start-Up, [2] Physical Grounding, [3] Experiment. These stages are described below along with the LLM prompts.

Phase 1: Experiment Start-Up: The goal of ORGANA.REASONER during the Experiment Start-Up phase is to gather enough information in order to be able to execute the experiment. ORGANA.REASONER has a series of questions that need to be answered in order to perform an experiment. ORGANA.REASONER asks these questions to a user and deduces the answers based on their responses according to the template below:

```
start_up_prompt = """
You are a robot chemist. A chemist will
ask you to perform an experiment. Your
goal is to find out all the information
```

*https://github.com/Uberi/speech_recognition
†https://github.com/openai/whisper
‡https://openai.com/gpt-4
§https://elevenlabs.io

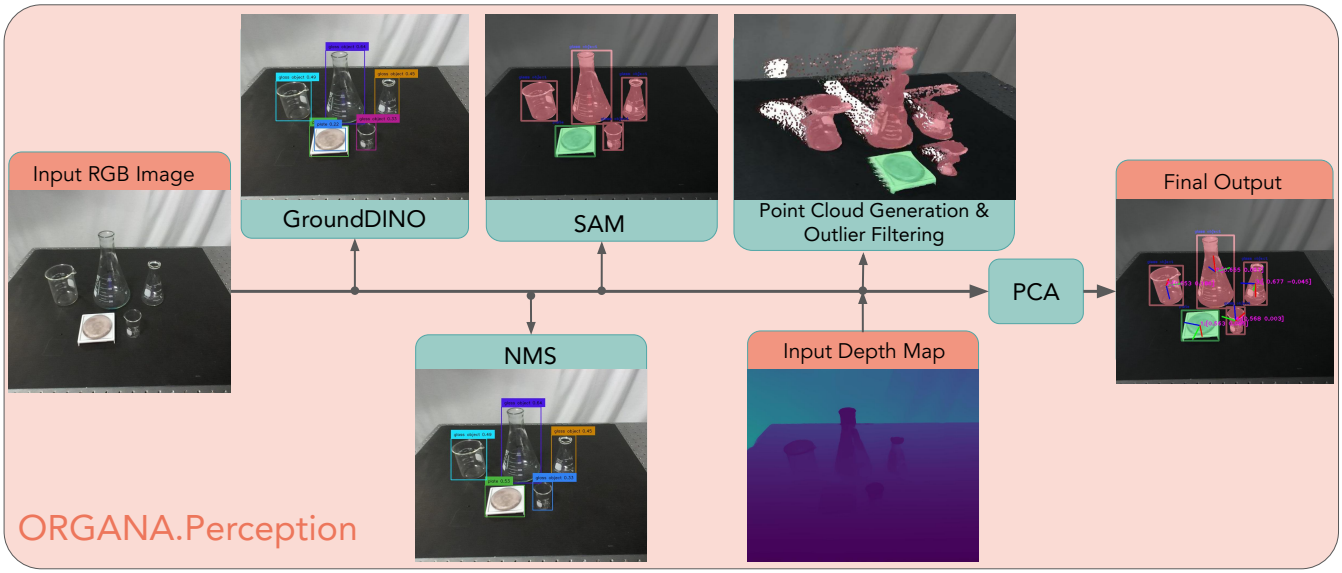


Fig. 12: **ORGANA Perception Pipeline.** Given an input RGB image, GroundDINO detects objects of interest based on pre-defined prompts and predicts 2D bounding boxes. NMS is then applied to remove redundant predictions, followed by SAM generating a segmentation mask for each detected object. To estimate object pose, the input depth map is used to generate 3D point clouds for each object, which are then filtered to remove outlier points. The object position is estimated based on the smallest 3D bounding box fitted to each set of point clouds, and the object rotation is estimated using PCA.

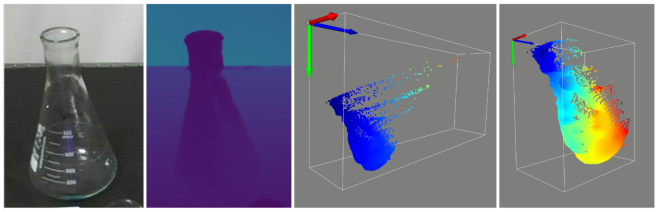


Fig. 13: **Visualizations of the RGBD image and 3D point clouds for an Erlenmeyer flask.** the two figures on the right display the point cloud generated from the raw sensor depth map, and the point clouds after outlier removal.

```
you need about the experiment. You fill
out the following items:::

[Item 1] What the experiment is:::
[Item 2] The setup of the lab
environment (what hardware and reagents
are present):::
[3] An example of how to run the
experiment
[Item 3a] The rationale behind the
example experiment:::
[Item 3b] How you would do the example
experiment:::
[Item 3c] What the expected outcome of
the example experiment is:::
[Item 4] Number of experiments to be run
 in total:::

You will receive input from the user.
You NEED to return two things:
[Item X] where X is the number of the
item satisfied or [Item None] if no
items were satisfied
[Question] The question you want to ask
next. You must only as one question at a
 time. If all the items have information
, write <DONE>.
"""
```

ORGANA.REASONER will prompt the chemist until all questions are satisfied. Below is an example of a completed prompt:

```
start_up_prompt = """
You are a robot chemist. A chemist will
ask you to perform an experiment. Your
goal is to find out all the information
you need about the experiment. You fill
out the following items:::

[Item 1] What the experiment is:::I
would like to run a chemistry
experimnetI would like to generate a
pourbaix plot for an unknown quinone
[Item 2] The setup of the lab
environment (what hardware and reagents
are present):::I have a solution
containing 10mM of the quinone, water, 1
M sodium chloride solution, and a series
 of 0.5 M pH X buffer solution (X is an
integer between 4 to 9)two beakers (
experiment beaker, waste beaker), a pH
probe and potentiostat
[3] An example of how to run the
experiment
[Item 3a] The rationale behind the
example experiment:::I want to measure
```

```
the potential of the quinone solution at
 various pHs. I will start at pH 4 and
take a measurement there. The
experiments should measure the potential
 at all pHs from 4 to 9.
[Item 3b] How you would do the example
experiment:::Add 2 mL of pH 4 buffer
solution. Add 1 ml NaCl solution. add 5
mL of water followed by 2 mL of quinone
solution. Then measure the pH and run a
CV scan. transfer r contents of beaker
to waste beaker
[Item 3c] What the expected outcome of
the example experiment is:::The measured
 pH should be around 4 (a little
difference is okay). Unsure what the
potential should be since it's the first
 experiment, but the trend is that
potential should become more negative as
 pH goes up.
[Item 4] Number of experiments to be run
 in total:::6

You will receive input from the user.
You NEED to return two things:
[Item X] where X is the number of the
item satisfied or [Item None] if no
items were satisfied
[Question] The question you want to ask
next. You must only ask one question at
a time. If all the items have
information, write <DONE>.
"""
```

The answers are processed into a dictionary of initial conditions:

```
init_conditions_dict = {"goal": ... #[
Item 1] from start_up_prompt,
                        "setup": ... #[
Item 2] from start_up_prompt,
                        "thought": ... #
[Item 3a] from start_up_prompt,
                        "action": ... #[
Item 3b] from start_up_prompt,
                        "expected_obs":
...#[Item 3c] from start_up_prompt,
                        "num_repeats":
... #[Item 4] from start_up_prompt
                        }
```

Based on the answers to the prompts, ORGANA.REASONER will then decide which category of experiment is being requested. This is necessary to determining post-processing functions for experiment results.

```
experiment_type = """
Based on this experiment goal: {0}, what
 type of experiment is being done?
Select type from [POURBAIX, TITRATION,
SOLUBILITY, RECRYSTALLIZATION, NONE].
Only return type, nothing else. If you
are uncertain, return NONE
"""
```

If the experiment type is not known, ORGANA will exit.

Phase 2: Physical Grounding: If the experiment category is valid, ORGANA.REASONER will enter the grounding phase, where ORGANA.REASONER shows the users vessels that it perceived and the user is asked to assign semantic meaning to the vessel so that it knows the vessel's role in the experiment using a graphical user interface (see Fig. 14).

```
grounding_question = """
What will this be used for?
"""
```

The user enters an answer in natural text for each vessel, which is assigned to the vessel. ORGANA.REASONER is called to match the name of the vessel that the user entered to a list of names it has in its knowledge base:

```
grounding_perception_prompt="""
These are the official names of hardware
 in a chemistry lab:
  {0}

These are user descriptions of those
hardware, not necessarily that order:
  {1}

Match the official names to the user
descriptions using the following format:
[start matching]
<user description>|<official_name>
[end matching]
"""
```

In the above prompt, {0} refers to the list of vessel in the knowledge base and {1} refers to the names input by a human. A sample grounding prompt is:

```
grounding_perception_prompt="""
These are the official names of hardware
 in a chemistry lab:
  ["washing_beaker", "experiment_beaker
", "pH_beaker"]

These are user descriptions of those
hardware, not necessarily that order:
  ["this beaker is used for washing", "
main exp beaker", "beaker for measuring
pH"]

Match the official names to the user
descriptions using the following format:
[start matching]
<user description>|<official_name>
[end matching]
"""
```

Phase 3: Experiment: To generate an experiment plan, ORGANA.REASONER is given the following prompt:

```
experiment_prompt="""
Experiment goal: {0}
Lab equipment and reagents: {1}

I will tell you how to run a single
experiment:

<Thought>{2}</Thought>
<Action>{3}</Action>
<Expected Observation>{4}</Expected
Observation>
```

```
These are past experiments that you did:

{5}

Can you propose the next experiment to
try, as well as the output you expect to
 see?

This was feedback receieved during the
last experiment, incorporate it when
planning your next experiment:

<Human Feedback>{6}</Human Feedback>

Rules:
1. Format the plan using <Thought>, <
Action>, <Expected Output> tags.
2. If you see feedback in <System
Feedback> tags, consider them when
planning the next experiment.
"""
```

{0} is the experiment goal taken from the init_conditions_dict['goal']. {1} is the list of lab equipment from init_conditions_dict['setup']. {2-4} are taken from the example provided by the user in init_conditions_dict['thought'], init_conditions_dict['action'], init_conditions_dict['observation']. After the first experiment, {5} is a memory prompt of past experiments performed. This is important so that ORGANA.REASONER does not propose the same experiment multiple times and is further explained in "Memories of past experiments". {6} is any feedback returned from humans about the experiment. Humans are asked for feedback if the results of the experiment do not match the expectations. This is further described in "Examples of ambiguity and uncertainty mitigation" below.

Memories of past experiments: To reduce the number of tokens used for memories, past observations are summarized for every k experiment. We use $k = 3$. The memory prompt is structured as follows:

```
memory_prompt="""
These are past experiments that you did:

This is a summary of what happened
during the first three experiments:

{0}

This is a summary of what happened
during the next three experiments:

{1}

...

This is what happened during the most
recent experiments:

<Thought>prev_exp['thought']</Thought>
<Action>prev_exp['action']</Action>
<Observation>prev_exp['observation']</
Observation>
"""
```

Every three experiments, memories are summarized using the following prompt:

```
memory_summary="""
These are experiments that a robot did:

{0}

Can you provide a summary of these
experiments?
"""
```

Once ORGANA.REASONER generates a new experiment plan, it is converted to XDL using CLAIRify. The robot performs the experiment and the results are sent back to ORGANA.REASONER for rationalization.

Once all the experiments are completed, ORGANA.REASONER is asked to generate a single final summary for the report:

```
final_summary="""
Below are summaries from a series of
experiments the robot did:

{0}

Provide a single summary for all these
experiments.
"""
```

Examples of ambiguity and uncertainty mitigation: ORGANA.REASONER is asked to rationalize about whether the outcomes of the experiment match user expectations. This is done using the following prompt:

```
rationalize_prompt="""
This was an experiment a robot was asked
 to do:

{0}

After performing the experiment, these
are the outcomes:

{1}

Do the actual outcomes of the experiment
 match the expected outcomes? Begin your
 answer with <YES> or <NO>, followed by
the explanation.
"""
```

If ORGANA.REASONER cannot rationalize is experiment, the human is pinged to determine if there are any issues in the environment. Below is an example of an instance where the experiment results did not match the user expectations:

```
# rationalization_output
"""
<NO>  The actual outcomes of the
experiment do not match the expected
outcomes. The pH measured at 5.96 is
close to the expected pH of 6, which is
within the acceptable range. However,
the potential at this pH level is
-0.492, which is not more negative than
the previous measurement at pH 4.83
(-0.681). This contradicts the expected
```

```
trend of the potential becoming more
negative as the pH increases.

pinging
human......

Observations don't make sense. Any
feedback on the experiment?
"""
```

The user then provides feedback on what (if anything) explains the results. ORGANA.REASONER can then incorporate that feedback into the next plan (for example, the user might say 'I'm not sure what happened, but repeat the previous experiment just in case', 'Nothing is wrong, carry on', 'The pump was stuck, it's ok again').

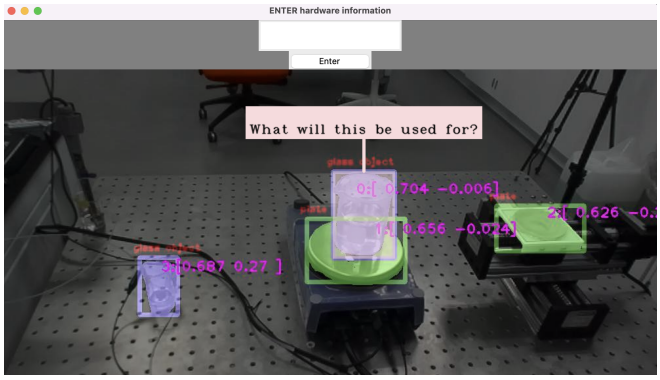


Fig. 14: **Visualization of the user interface for grounding perception.** Users can respond by typing in the text box or simply talking to ORGANA.

C. User Study

User Study Participants: We recruited 8 chemists to test out ORGANA. The expertise of the chemists ranged from novice (no experimental background) to expert (13 years of experience). 25% of participants identified as female and 75% as male. Ages ranged from early twenties to early thirties.

SUS questionnaire: The SUS questionnaire was given to users both after performing the manual experiment and after using ORGANA.REASONER. The SUS contains the following statements, which users rated on a Likert scale from 1 (strongly disagree) to 5 (strongly agree) [69].

1) I think that I would like to use this system frequently.
2) I found the system unnecessarily complex.
3) I thought the system was easy to use.
4) I think that I would need the support of a technical person to be able to use this system.
5) I found the various functions in this system were well integrated.
6) I thought there was too much inconsistency in this system.
7) I would imagine that most people would learn to use this system very quickly.
8) I found the system very cumbersome to use.
9) I felt very confident using the system.
10) I needed to learn a lot of things before I could get going with this system.

User ratings were transformed to a score out of 10 for each question, where higher is better. The formula for the score s_i for each question x_i is:

- $s_i = 2.5 * (x_i - 1)$ for $x_i \in \{1, 3, 5, 9\}$
- $s_i = 2.5 * (5 - x_i)$ for $x_i \in \{2, 4, 6, 8, 10\}$

NASA Task Load Index (TLX) questions: We gave users the NASA-TLX quesetionnaire [68] both after performing the electrochemistry experiment manually and after using ORGANA to perform the experiment. The questions are listed below, and the user grades each from 0-20. The score is divided by two to get a score out of 10.

1) Mental demand: How mentally demanding was the task? (0: very low, 20: very high)
2) Physical demand: How physically demanding was the task? (0: very low, 20: very high)
3) Temporal demand: How hurried or rushed was the pace of the task? (0: very low, 20: very high)
4) Performance: How successful were you in accomplishing what you were asked to do? (0: perfect, 20: failure)
5) Effort: How hard did you have to work to accomplish your level of performance? (0: very low, 20: very high)
6) Frustration: How insecure, discouraged, irritated, stressed, and annoyed were you? (0: very low, 20: very high)

D. Hardware for Perception

We integrated diverse hardware components for conducting a range of chemistry measurements. Specifically, an IKA RET control-visc served as a weighing scale and was utilized for pH measurements by interfacing with a pH probe (Orion ROSS Ultra Refillable pH/ATC Triode Combination Electrodes for Orion Series Meters, Thermo Fisher Scientific). Additionally, a Sartorius BCA2202-1S Entris functioned as a high-precision scale. Electrochemical properties of solutions were assessed using a portable potentiostat [61], employing cyclic voltammetry measurements. Two cameras were used in our experimental setup. The Intel RealSense D435i stereo camera, mounted on the robot end-effector, not only estimated object poses in the scene by detecting fiducial markers through the AprilTag library[82] but also facilitated the measurement of solution turbidity. Given the prevalence of transparent objects in the laboratory, as highlighted in [43], we employed a ZED Mini camera for transparent object detection and depth estimation. We utilized the ZED camera depth map in neural depth mode. The hand-eye calibration for both cameras was executed using an infrastructure robotics library[¶]. These devices were connected to the robot controller machine via USB protocol.

a) Redox potential feedback: We ran a cyclic voltammetry (CV) measurement [87] to measure the redox potential of the quinone solution at a given pH. A portable potentiostat [61] was connected to the robot controller machine

[¶]https://github.com/uoft-cs-robotics/robot_system_tools

and CV measurement was conducted with a standard 3-electrode system, where the working electrode was a glassy carbon, and the counter and reference electrodes were silver wires. The reduction and oxidation peaks in the CV plot were automatically detected by picking up the minimum and maximum values in the measured voltage range. The mean voltage of the two peak voltages was used as the redox potential of the sample.

E. Hardware for Actions

ORGANA incorporates several hardware solutions for diverse chemistry experiments. For solution heating and stirring, we employed the IKA RET control-visc, interfacing seamlessly with the robot workstation via PyVISA [88]. In our electrochemistry experiment, ORGANA utilizes a polishing station to refine the glassy carbon electrode. This station comprises a polishing pad connected to two linear actuators that execute planar motion. A mechanical impedance, facilitated by a spring linked to the electrode jig, governs the normal interaction force between the polishing station and the robot end-effector. Further details on the polishing process are available in [65]. For precise liquid transfers between different stations in the electrochemistry experiment, we employed a Cavro XCalibur Pump featuring a 12-port ceramic valve (Tecan Systems). This pump, connected to 12 reagent vials through tubes, accurately conveyed reagents into the specified container upon request. Moreover, for all the experiments a Franka Emika Panda arm robot, equipped with a Robotiq 2F-85 gripper was used. To facilitate grasping objects from the side in tabletop scenarios and enhance constrained motion planning, we positioned the end-effector parallel to the ground on the robot's last link, as depicted in Figure 5. This configuration was attained either through a fixed linkage or by incorporating a Dynamixel XM540-W150 servo motor as an additional degree of freedom.

F. Electrochemistry Parameter Estimation

In the electrochemisty experiment we have two goals for parameter estimation. First, to produce a maximum likelihood estimate for the model parameters given the data, values which we use as the output of the experiment. Second, to estimate the posterior distribution in the parameter space given the data and individual marginal distribution for each parameter, which we use to give visual representation of the progress of the experiment to the chemist.

Maximum likelihood estimate. We assume that our data points are coming from a Gaussian distribution with a mean given by the model described in Section III-E.1. We assume that the variance does not depend on pH, but do not assume that it is known and simply add it as an additional parameter to the estimation. Our model is therefore given with:

$$\theta = (\mathrm{pK_{A1}}, \mathrm{pK_{A2}}, \mathrm{k}, \mathrm{E_{inf}}, \sigma_{\mathrm{eV}}) \tag{2}$$

$$\mu_{\mathrm{eV}}(\mathrm{pH}) = \begin{cases} \mathrm{E_{inf}} - \mathrm{k}(\mathrm{pK_{A2}} - \mathrm{pK_{A1}}) - 2\mathrm{k}(\mathrm{pK_{A1}} - \mathrm{pH}), \\ \quad \mathrm{pH} < \mathrm{pK_{A1}} \\ \mathrm{E_{inf}} - \mathrm{k}(\mathrm{pK_{A2}} - \mathrm{pH}), \\ \quad \mathrm{pK_{A1}} \leq \mathrm{pH} \leq \mathrm{pK_{A2}} \\ \mathrm{E_{inf}}, \\ \quad \mathrm{pK_{A2}} < \mathrm{pH} \end{cases} \tag{3}$$

$$\sigma_{\mathrm{eV}}(\mathrm{pH}) = \sigma_{\mathrm{eV}} \tag{4}$$

$$eV(pH) \sim \mathcal{N}(\mu_{\mathrm{eV}},\, \sigma^2_{\mathrm{eV}}) \tag{5}$$

Likelihood for a set of data points $\{(\mathrm{pH}, \mathrm{eV})_i\}$ given model parameters is then:

$$p(\{(\mathrm{pH}, \mathrm{eV})_i\} \mid \theta) = \prod_i \frac{1}{\sigma^\theta_{\mathrm{pH}}(\mathrm{pH}_i)\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\mathrm{eV}_i - \mu^\theta(\mathrm{pH})}{\sigma^\theta(\mathrm{pH})}\right)^2\right) \tag{6}$$

This is all we need to produce the maximum likelihood estimate. We use `fmin` method from `scipy.optimize` Python package to produce the estimate. We use these estimated parameter values in each place where we report numeric results for this experiment in the paper as well as in the example report produced by the experiment (Appendix VIII-G).

Posterior distribution. We additionally aim to produce an estimate of the entire posterior distribution over the parameter space. In order to do this we assume uniform prior over the parameters. From there we can express the posterior as:

$$p(\theta \mid \{\mathrm{pH}_i, \mathrm{eV}_i\}) = \frac{p(\{\mathrm{pH}_i, \mathrm{eV}_i\} \mid \theta) p(\theta)}{p(\{\mathrm{pH}_i, \mathrm{eV}_i\})} \tag{7}$$

With uniform prior, up to a normalization constant we have:

$$p(\theta \mid \{\mathrm{pH}_i, \mathrm{eV}_i\}) \propto p(\{\mathrm{pH}_i, \mathrm{eV}_i\} \mid \theta) \tag{8}$$

$$p(\theta \mid \{\mathrm{pH}_i, \mathrm{eV}_i\}) \propto \prod_i \frac{1}{\sigma^\theta_{\mathrm{pH}}(\mathrm{pH}_i)\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\mathrm{eV}_i - \mu^\theta(\mathrm{pH})}{\sigma^\theta(\mathrm{pH})}\right)^2\right) \tag{9}$$

In other words, we are simply aiming to produce a normalized likelihood function over the space of parameters.

Based on this, we can now estimate marginal posterior distribution for each parameter (shown in individual parameter distribution plots in Figure 6 and Figure 8). We do this using importance sampling, utilizing samples from a uniform distribution covering the possible range for each parameter. We can express the marginal distribution as (given

for $p(\text{pK}_{\text{A1}})$ here, same for all other parameters):

$$p(\text{pK}_{\text{A1}} \mid \{\text{pH}_i, \text{eV}_i\}) = \int_\theta \text{pK}_{\text{A1}}^\theta p(\theta \mid \{\text{pH}_i, \text{eV}_i\}) d\theta \quad (10)$$

$$p(\text{pK}_{\text{A1}} \mid \{\text{pH}_i, \text{eV}_i\}))$$
$$\approx \frac{1}{N} \sum_{j=1}^{N} \frac{p(\theta \mid \{\text{pH}_i, \text{eV}_i\})}{q(\theta)} \text{pK}_{\text{A1}}^\theta, \text{ where } \theta \sim q. \quad (11)$$

$$p(\text{pK}_{\text{A1}} \mid \{\text{pH}_i, \text{eV}_i\})$$
$$\approx \frac{1}{S} \sum_{j=1}^{N} p(\{\text{pH}_i, \text{eV}_i\} \mid \theta) \text{pK}_{\text{A1}}^\theta,$$
where $\theta \sim Unif$and S is a normalization constant. (12)

Finally, we can give the estimate for the distribution of the mean values for possible models at each pH (essentially a distribution over possible model lines). We use this to visually represent our current belief over possible models and give the user an easily digestible overview of the current status of the experiment (top left model line distribution plots in Figure 6 and Figure 8).

$$p(\mu_{\text{eV}}(\text{pH}) \mid \{\text{pH}_i, \text{eV}_i\})$$
$$= \int_\theta \mu_{\text{eV}}(\text{pH} \mid \theta) p(\theta \mid \{\text{pH}_i, \text{eV}_i\}) d\theta \quad (13)$$

$$p(\mu_{\text{eV}}(\text{pH}) \mid \{\text{pH}_i, \text{eV}_i\})$$
$$\approx \frac{1}{N} \sum_{j=1}^{N} \frac{p(\theta \mid \{\text{pH}_i, \text{eV}_i\})}{q(\theta)} \mu_{\text{eV}}(\text{pH} \mid \theta), \quad (14)$$
where $\theta \sim q$.

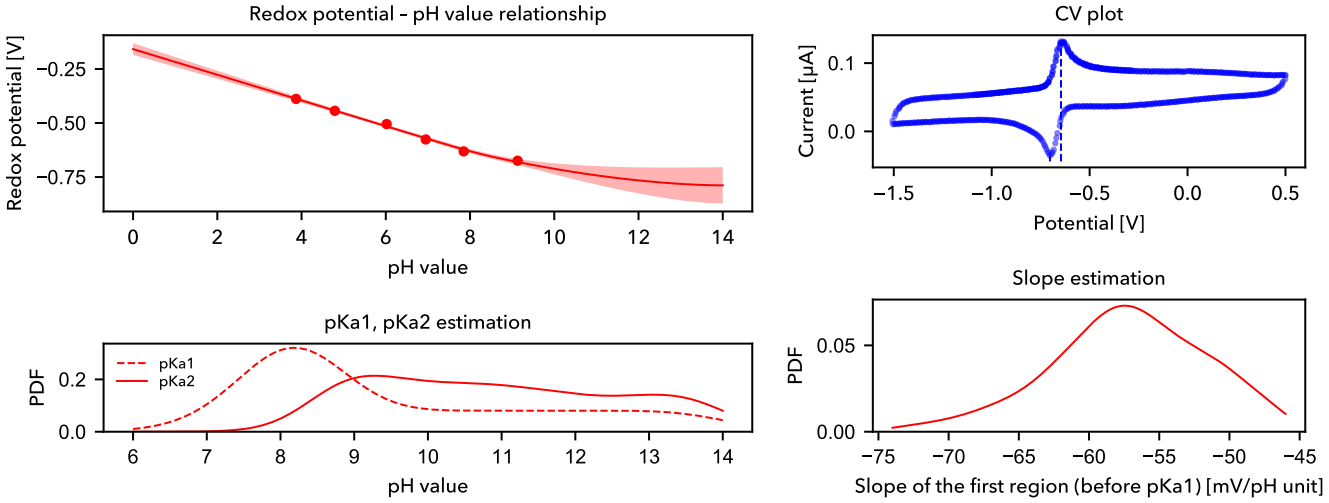
$$p(\mu_{\text{eV}}(\text{pH}) \mid \{\text{pH}_i, \text{eV}_i\})$$
$$\approx \frac{1}{S} \sum_{j=1}^{N} p(\{\text{pH}_i, \text{eV}_i\} \mid \theta) \mu_{\text{eV}}(\text{pH} \mid \theta),$$
where $\theta \sim Unif$and S is a normalization constant. (15)

G. Analysis Report

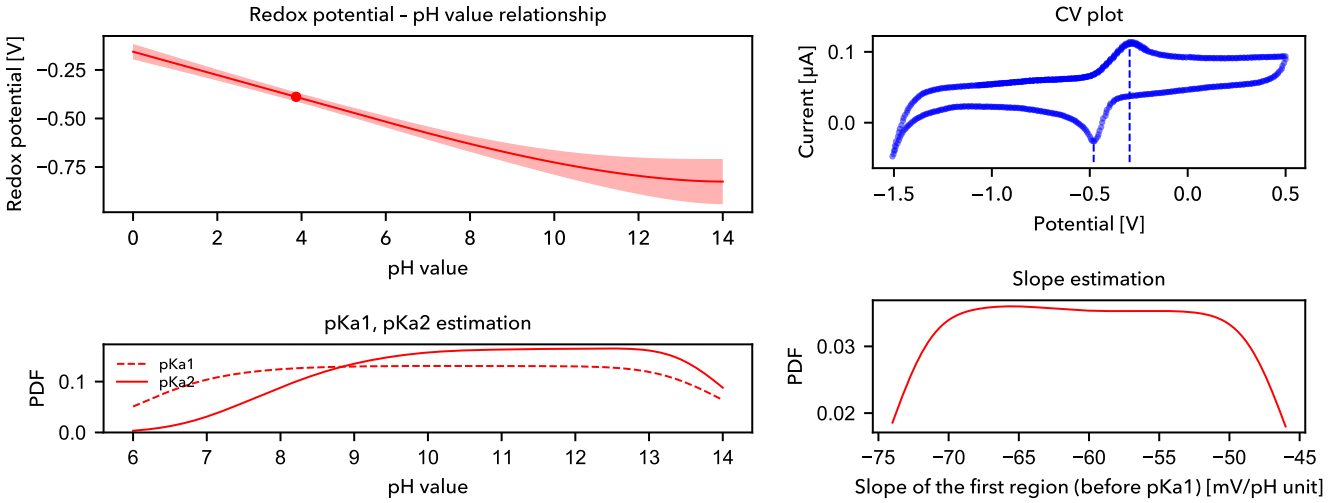
We present an example report generated by the ORGANA for electrochemistry experiments aimed at obtaining the Pourbaix plot.

Note: One might notice a discrepancy between the estimated value given for one of the parameters and the location of the highest probability in the corresponding plot. The plots represent the marginal distribution for each parameter. They are mainly being used as a visualization tool, to give the chemist an overview of the current progress of the experiment. On the other hand, the given estimate values represent the set of parameters with the maximum likelihood and are what we report as the result of the experiment.

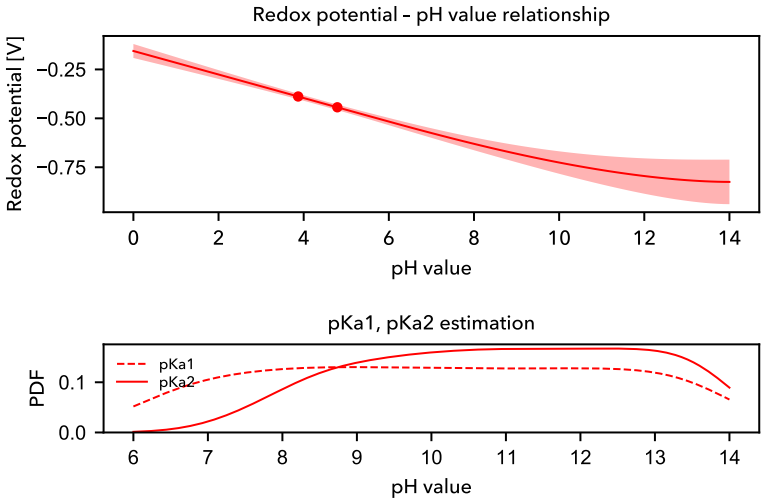
```
The series of experiments were conducted to measure the potential of a quinone
solution at various pH levels, specifically from pH 7 to pH 9. The procedure
involved adding 6 mL of buffer solution of the desired pH, 3 mL of NaCl solution, 15
 mL of water, and 6 mL of quinone solution to a beaker. A CV scan was then run to
measure the potential. The contents of the beaker were then transferred to a waste
beaker. The observations showed that as the pH increased, the potential became more
negative. The potential at pH 3.87 was -0.388 eV, at pH 4.79 was -0.443 eV, at pH
6.02 was -0.505 eV, at pH 6.95 was -0.576 eV, at pH 7.85 was -0.631 eV, and at pH
9.13 was -0.674 eV.
After performing the experiments, these are the results:

The estimate for pKa1 is 8.096.
The estimate for pKa2 is 12.380.
The estimate for slope is -60.958.
```

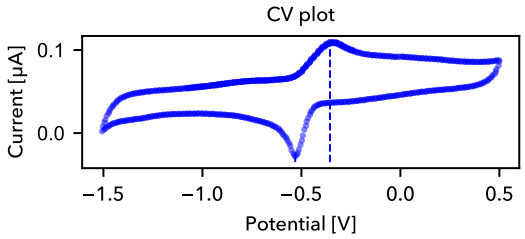


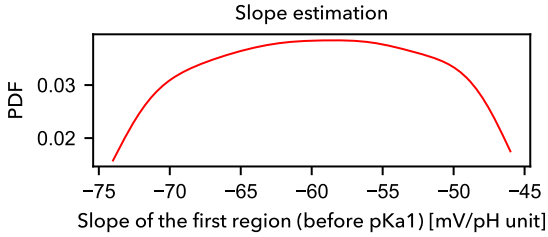
```
.....................Log for experiment 1.....................


This was the rationalization behind the experiment:
I want to measure the potential of the quinone solution at various pHs. I will start
 at pH 4 and take a measurement there. The experiments should measure the potential
at all pHs from 4 to 9.

This was the experiment protocol that was done:
Add 6 mL of pH 4 buffer solution. Add 3 ml NaCl solution. add 15 mL of water
followed by 6 mL of quinone solution. Then measure the pH and run a CV scan.
transfer contents of beaker to waste beaker.

This was the expected output from the experiment:
The measured pH should be around 4 (a little difference is okay). Unsure what the
potential should be since it's the first experiment, but the trend is that potential
 should become more negative as pH goes up.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}]


After performing the experiments, these are the results:

The estimate for pKa1 is 6.047.
The estimate for pKa2 is 7.682.
The estimate for slope is -145.438.
```



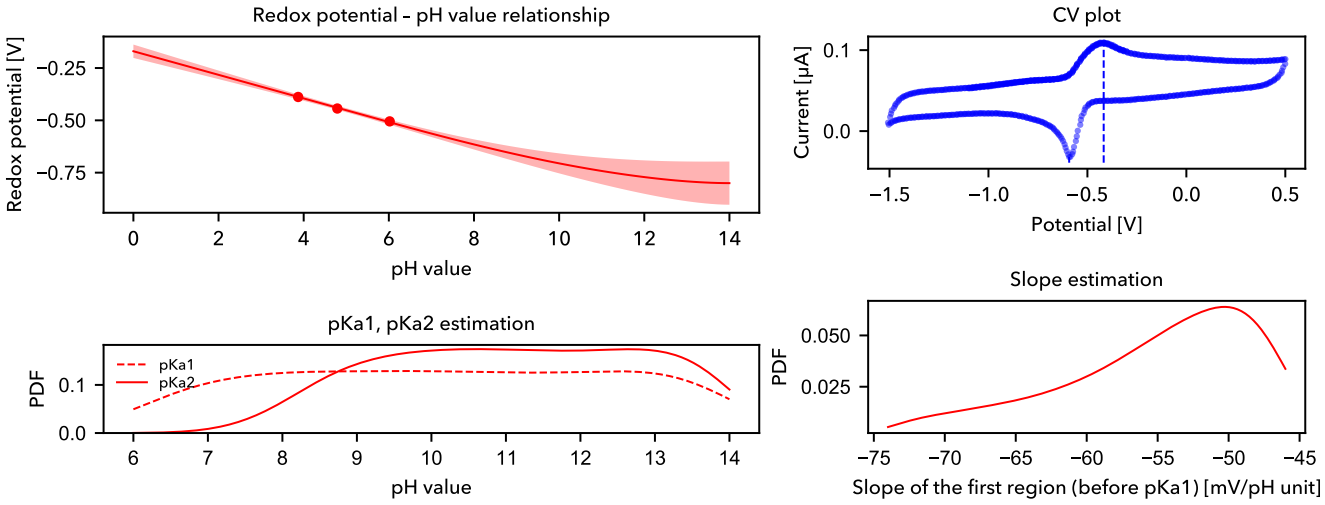
```
.....................Log for experiment 2.....................


This was the rationalization behind the experiment:
Based on the previous experiment, the pH was slightly lower than expected, but
within an acceptable range. The potential was also measured. Now, I will proceed to
the next pH level, which is pH 5, and measure the potential there. I expect the
potential to become more negative as the pH increases.

This was the experiment protocol that was done:
Add 6 mL of pH 5 buffer solution to the experiment beaker. Add 3 ml NaCl solution.
Add 15 mL of water followed by 6 mL of quinone solution. Then measure the pH and run
 a CV scan. Transfer contents of beaker to the waste beaker.

This was the expected output from the experiment:
The measured pH should be around 5 (a little difference is okay). The potential
should be more negative than the previous measurement at pH 4.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}, {'pH': 4.79, 'eV': -0.4434841666666667}]


After performing the experiments, these are the results:

The estimate for pKa1 is 10.707.
The estimate for pKa2 is 13.185.
The estimate for slope is -59.963.
```



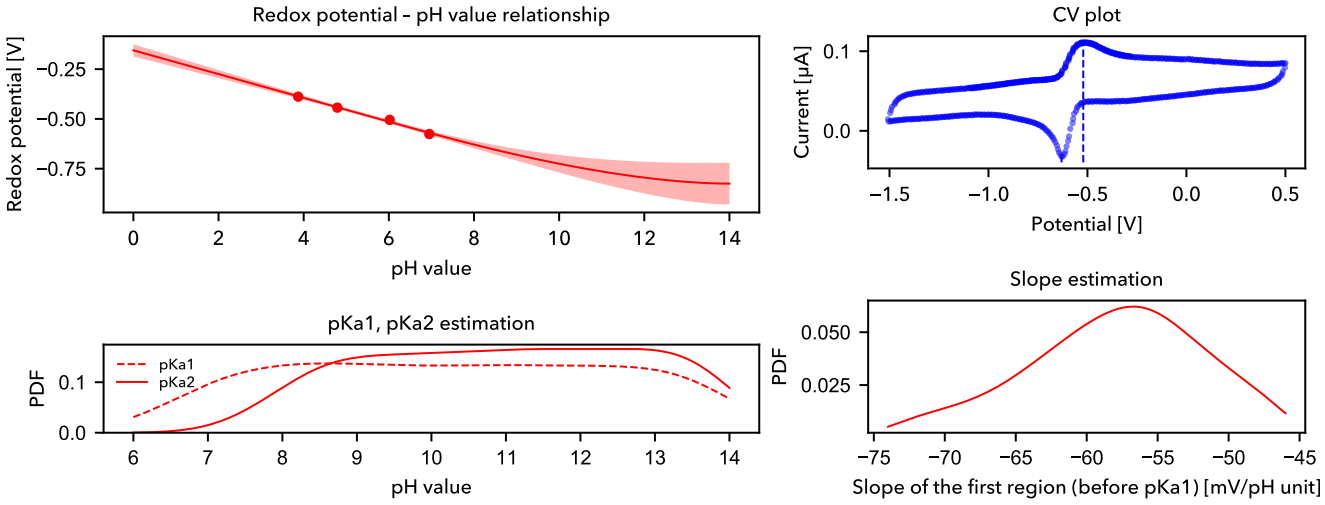




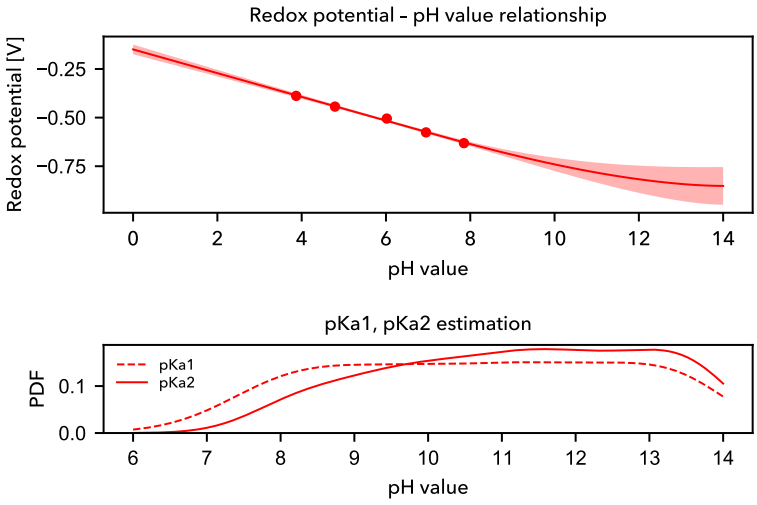
```
.....................Log for experiment 3.....................


This was the rationalization behind the experiment:
Based on the previous experiments, the pH was slightly lower than expected, but
within an acceptable range. The potential was also measured. Now, I will proceed to
the next pH level, which is pH 6, and measure the potential there. I expect the
potential to become more negative as the pH increases.

This was the experiment protocol that was done:
Add 6 mL of pH 6 buffer solution to the experiment beaker. Add 3 ml NaCl solution.
Add 15 mL of water followed by 6 mL of quinone solution. Then measure the pH and run
 a CV scan. Transfer contents of beaker to the waste beaker.

This was the expected output from the experiment:
The measured pH should be around 6 (a little difference is okay). The potential
should be more negative than the previous measurement at pH 5.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}, {'pH': 4.79, 'eV': -0.4434841666666667},
{'pH': 6.02, 'eV': -0.5049748251748252}]


After performing the experiments, these are the results:

The estimate for pKa1 is 8.646.
The estimate for pKa2 is 18.752.
The estimate for slope is -54.026.
```

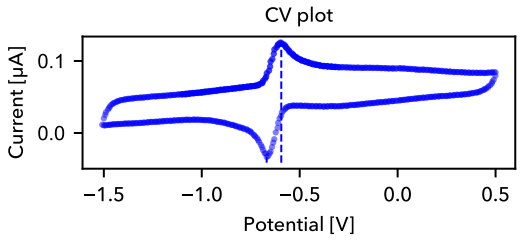


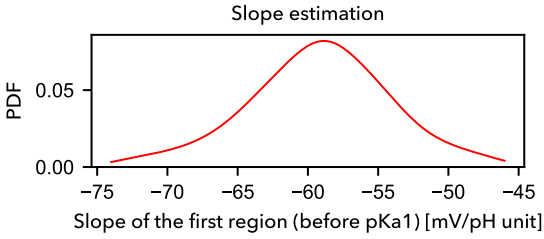
```
.....................Log for experiment 4.....................


This was the rationalization behind the experiment:
I want to measure the potential of the quinone solution at various pHs. I will start
 at pH 7 and take a measurement there. The experiments should measure the potential
at all pHs from 7 to 9.

This was the experiment protocol that was done:
Add 6 mL of pH 7 buffer solution. Add 3 ml NaCl solution. add 15 mL of water
followed by 6 mL of quinone solution. Then measure the pH and run a CV scan.
transfer contents of beaker to waste beaker.

This was the expected output from the experiment:
The measured pH should be around 7 (a little difference is okay). Unsure what the
potential should be since it's the first experiment, but the trend is that potential
 should become more negative as pH goes up.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}, {'pH': 4.79, 'eV': -0.4434841666666667},
{'pH': 6.02, 'eV': -0.5049748251748252}, {'pH': 6.95, 'eV': -0.57605}]

A human was asked to intervene in the experiment, and this was their feedback:
go


After performing the experiments, these are the results:

The estimate for pKa1 is 7.303.
The estimate for pKa2 is 19.501.
The estimate for slope is -59.451.
```



```
.....................Log for experiment 5.....................


This was the rationalization behind the experiment:
Based on the previous experiment, the potential became more negative as the pH
increased. I will continue this trend and measure the potential at pH 8.

This was the experiment protocol that was done:
Add 6 mL of pH 8 buffer solution to the experiment beaker. Add 3 ml NaCl solution.
Add 15 mL of water followed by 6 mL of quinone solution. Then measure the pH and run
 a CV scan. After the measurement, transfer the contents of the experiment beaker to
 the waste beaker.

This was the expected output from the experiment:
The measured pH should be around 8. The potential should be more negative than the
previous measurement at pH 7.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}, {'pH': 4.79, 'eV': -0.4434841666666667},
{'pH': 6.02, 'eV': -0.5049748251748252}, {'pH': 6.95, 'eV': -0.57605}, {'pH': 7.85,
'eV': -0.6313348484848484}]


After performing the experiments, these are the results:

The estimate for pKa1 is 7.945.
The estimate for pKa2 is 17.386.
The estimate for slope is -60.958.
```







```
.....................Log for experiment 6.....................


This was the rationalization behind the experiment:
Based on the previous experiments, the potential became more negative as the pH
increased. I will continue this trend and measure the potential at pH 9, which is
the final pH level in the series.

This was the experiment protocol that was done:
Add 6 mL of pH 9 buffer solution to the experiment beaker. Add 3 ml NaCl solution.
Add 15 mL of water followed by 6 mL of quinone solution. Then measure the pH and run
 a CV scan. After the measurement, transfer the contents of the experiment beaker to
 the waste beaker.

This was the expected output from the experiment:
Based on the trend observed in previous experiments, the measured pH should be
around 9 (a little difference is okay) and the potential should be more negative
than the previous measurement at pH 8.

This was the actual output from the experiment:
[{'pH': 3.87, 'eV': -0.3883181818181818}, {'pH': 4.79, 'eV': -0.4434841666666667},
{'pH': 6.02, 'eV': -0.5049748251748252}, {'pH': 6.95, 'eV': -0.57605}, {'pH': 7.85,
'eV': -0.6313348484848484}, {'pH': 9.13, 'eV': -0.6744594444444445}]


After performing the experiments, these are the results:

The estimate for pKa1 is 8.096.
The estimate for pKa2 is 12.380.
The estimate for slope is -60.958.
```