Journal Name

ARTICLE TYPE

Cite this: DOI: 00.0000/xxxxxxxxxx

PRISM：通过智能仿真建模进行协议改进†

Brian Hsu [b]、Priyanka V Setty [†a,b]、Rory M Butler [†a,b]、Ryan Lewis [a]、Casey Stone [a]、Rebecca Weinberg [a]、Thomas Brettin [a,b]、Rick Stevens [a,b]、Ian Foster [b]、Arvind Ramanathan [a*]

Received Date
Accepted Date

DOI: 00.0000/xxxxxxxxxx

arXiv:2601.05356v1 [cs.RO] 8 Jan 2026

实验方案设计与执行的自动化，仍然是实现自驱动实验室的一个根本性瓶颈。我们提出了PRISM（通过智能仿真建模进行协议改进），这是一个能够在由现成的机器人仪器组成的实验室平台上，自动完成实验方案设计、验证和执行的框架。PRISM 使用一组基于语言模型的智能体，协同生成并优化实验步骤。该过程首先从描述实验工作流程的网络源中自动收集相关操作流程。随后，通过规划、Critique 和验证循环，将这些流程转换为结构化的实验步骤（例如液体处理步骤、平台布局及其他相关操作）。最终确定的步骤会被转换为 Argonne MADSci 协议格式；该格式为协调多种机器人仪器（Opentrons OT-2液体处理器、PF400 机械臂、Azenta 封板机和揭膜机）提供统一接口，且步骤之间无需人工干预。为评估实验方案生成性能，我们在受限式和开放式提示范式下，对单一推理模型和多智能体工作流进行了基准测试。生成的实验方案在 NVIDIA Omniverse 中构建的数字孪生环境内进行了验证，以便在执行前检测物理错误或顺序错误。我们以 Luna qPCR 扩增和细胞绘画为案例研究，展示了 PRISM 作为一种实用的端到端工作流程，如何贯通基于语言的实验方案生成、基于仿真的验证以及自动化机器人执行。

1 引言

实验方案设计与执行的自动化仍是实现自驱动实验室的一大根本瓶颈。尽管机器人平台日益普及，将科学意图转化为可执行的实验室方案仍需要丰富的领域专业知识，因为这一过程容易出错，且要求深入了解实验流程和仪器特定的格式要求。现有的协议描述语言提供了结构化表示，但仍然是静态且依赖特定硬件的；实验室配置发生变化时，必须手动重写。近期研究表明，大语言模型（LLM）能够根据自然语言描述生成看似合理的实验程序，但这些输出往往存在参数说明不足、物理上不可行或步骤顺序错误等问题，进而导致执行失败。与此同时，仿真和数字孪生技术推动了实验室监测和记录的发展，但尚未作为有效的执行前验证工具融入实验方案生成流程。直接在物理硬件上测试未经验证的实验方案代价高昂、耗时，而且有设备损坏和材料浪费的风险。这些挑战促使人们开发将自动化实验方案生成与严格的执行前验证相结合的系统，以实现安全、可靠且完全自主的实验室运行。

1.1 协议语言与实验室自动化标准

早期将实验室流程形式化的工作，侧重于创建实验步骤的机器可读描述。XDL [1]通过用一种硬件无关语言表达加热或添加试剂等操作，展示了这一方法在自动化化学中的应用。Autoprotocol [2]进一步将这一理念推广到更广泛的云端运行实验室，并引入一种基于 JSON 的规范，用于描述移液、孵育、封板和测量等通用实验室操作，使实验方案能够在各种自动化系统上执行。Aquarium [3]等集成度更高的平台，则将实验方案规范与库存跟踪和执行

[a] 阿贡国家实验室，*9700 S. Cass Avenue*，莱蒙特，*IL 60439*，美国。[b] 芝加哥大学计算机科学系，芝加哥，*IL 60637*，美国。
* 电子邮件：*ramanathana@anl.gov*
† 这些作者的贡献相同。

监督相结合，利用高级语言和实验室信息管理系统（LIMS）支持技术人员参与其中的工作流程。

尽管这些系统展现了实验朝着更加结构化、可重复的方向稳步发展，但它们在若干重要方面仍然存在局限。具体而言，实验方案描述是静态的，依赖特定配置，且几乎不支持在执行前验证物理可行性。此外，开发和调整这些方案还需要专业技术知识。这些不足促使我们探索更具适应性的方法，将结构化表示与上下文推理和自动验证相结合。

1.2 直接由大语言模型引导的实验方案生成

大语言模型与外部工具结合使用时，已展现出生成多步骤实验工作流程的能力。近期的综述梳理了跨学科的 260 多种此类“科学大语言模型”[4]，，并将它们归类为“自主智能体”：它们结合推理、规划和工具使用，以解决复杂的科学任务[5,6]。ChemCrow[7] 通过将大语言模型与反应数据库及规划器相结合，展示了大语言模型在化学工作流程中的应用，使高层目标能够分解为可执行的步骤。Coscientist[8] 则更进一步，整合了文档检索功能，以自主执行反应。

近期，ChemAgents[9] 和 LLM-RDF[10] 等分层多智能体系统已展示了端到端化学发现的能力，但值得注意的是，它们依赖人工参与机制或人工验证，以确保执行前的安全性。在此基础上，BioPlanner[11] 考察了类似能力能否扩展到生物实验方案，并评估模型生成流程的完整性和清晰度，开展了有限的可行性测试。类似地，Inagaki 等人[12] 和 Alchemist[13] 利用大语言模型生成可执行的机器人脚本（例如用于 Opentrons 的脚本），并证明虽然可以通过标准 API 模拟器捕获语法错误，但缺乏可靠环境模拟时，物理执行失败仍然存在。

而 ProtoCode[14]，等较新的研究则通过引入结构化中间表示，衔接自由文本描述与机器可解释的机器人操作，从而推动输出变得可执行。Agentic Lab[15] 采用多智能体编排进行知识检索和实验方案编排，并通过基于大语言模型的自我反思和人工反馈实现迭代优化。

这些努力共同反映出，人们越来越有兴趣利用语言模型，不仅勾勒实验意图，还指导下游自动化系统。然而，尽管取得了这些进展，仍存在一些局限。首先，参数规定不足很常见：模型输出往往遗漏执行所必需的关键信息，例如试剂用量、浓度、孵育时间或混合循环。其次，有些指令在物理上不可行，例如建议移动超出机械臂可达范围的实验室器具、以不符合平台布局的方式堆叠微孔板，或重新摆放现有硬件无法抓取的物品。事实上，近期针对实验室安全的基准测试发现，即使是当前最先进的模型，在超过 30% 的真实场景中也无法识别关键危害，这表明未经验证就执行存在风险{v1}。

第三，文本生成与机器人执行之间存在差距：大多数输出仍采用自然语言或半结构化伪代码，因此需要人工将其转换为实验室机器人使用的特定格式（例如 YAML 或 JSON）。最后，当前评估提供的验证或反馈有限，主要依赖专家审查，而非仿真或数字孪生测试。BioPlanner 包含小规模实验验证，但此类验证仍属例外。Agentic Lab 则通过增强现实眼镜和视觉语言模型，在人类进行物理操作时提供实时反馈。不过，这种方法是在执行过程中验证操作流程，而不是事先仿真可能的结果。

这些发现表明，大语言模型虽然能够协助将高层次的科学意图转化为操作步骤，但其输出仍需经过系统性的完善和物理验证。弥合这一差距，需要将语言生成与仿真、上下文推理和平台感知验证相结合——这正是本研究开发 PRISM 框架的动机。

1.3 仿真与数字孪生

数字孪生（在此语境下指物理实验室系统的虚拟副本）为设计、测试、部署、监控和对照现实世界中的机器人流程提供了机会[17]。这类仿真环境能够通过基于物理属性的碰撞检测、关节约束和逼真的运动规划，对机器人操作进行离线测试和验证。然而，大多数将数字孪生应用于实验室自动化的工作仅强调事后记录、执行追踪和实时监控，而非执行前验证与完善。催化领域近期的综述指出了这一差距，并提到尽管大语言模型具有灵活性，但利用数字孪生确保生成的实验方案具备物理安全性和完整性的工作仍处于起步阶段[18]。

大语言模型应用领域以往的进展展示了一些关键但有限的执行前验证与优化技术。例如，CLAIRIFY[19] 开发了一种利用验证器辅助迭代提示的方法，用于生成领域特定语言中的有效程序。通过将自动迭代提示与语法和基于规则的程序验证相结合，CLAIRIFY 确保生成的程序有效并纳入环境约束。然而，尽管这种验证会检查语法正确性和某些人工定义的规则，却无法通过高保真仿真全面验证物理可行性、碰撞风险或硬件约束。OR-GANA[20]，在 CLAIRIFY 的基础上，直接在物理硬件上执行实验方案，并在方案执行过程中即时解决任务与运动规划问题。同样，Agentic Lab 通过视觉语言模型进行实时监控，能够基于执行后分析持续学习并优化实验方案，但无法提供基于仿真的方法所具备的、有实际依据的执行前验证。这些系统检测和纠正错误的能力取决于

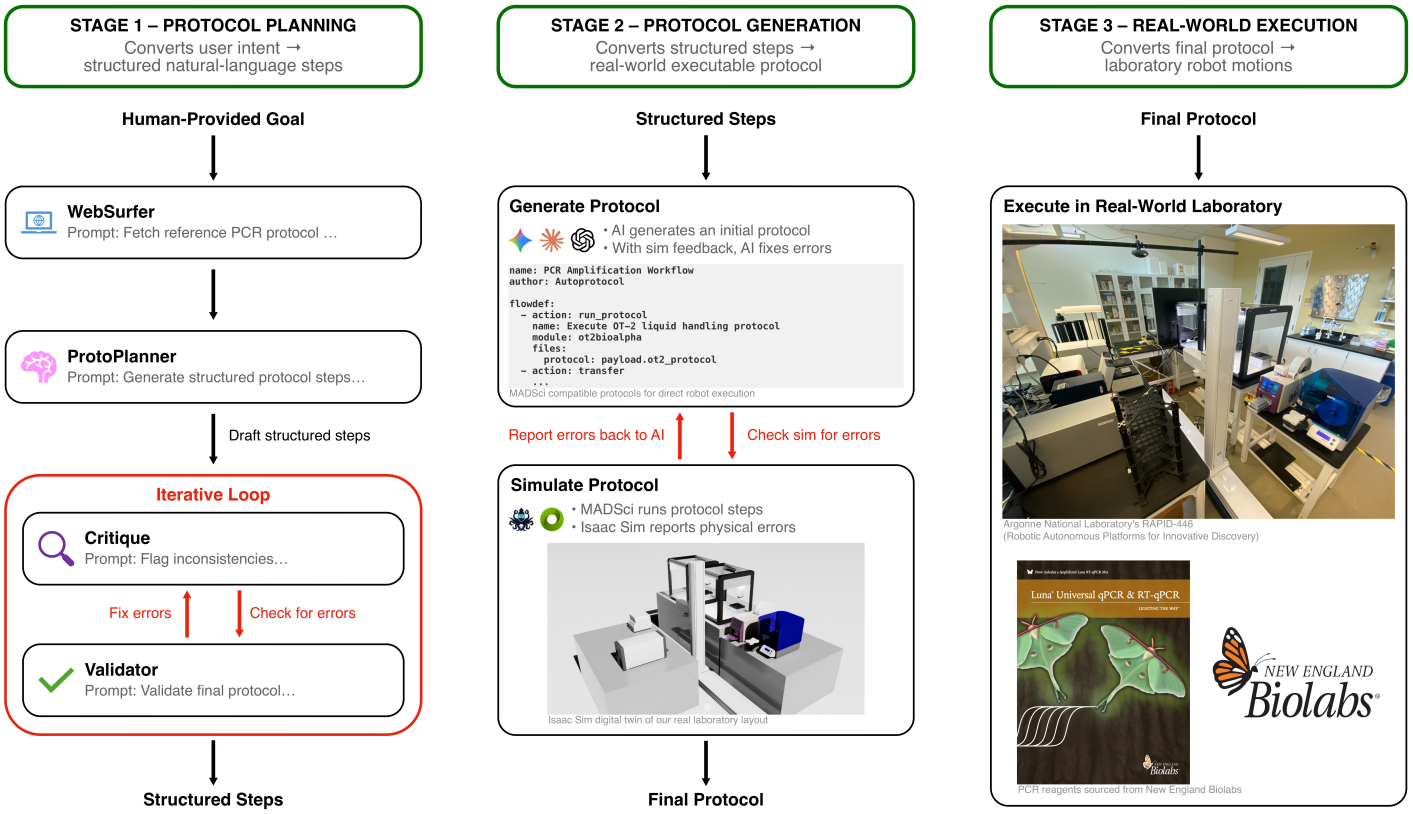


图 1 PRISM 框架用于实验方案生成与执行的概览。该系统由三个主要阶段组成：实验方案规划阶段，将用户意图转换为结构化步骤；实验方案生成阶段，将结构化英语指令转换为机器人感知动作，并在执行前通过 Omniverse 中的验证循环进行迭代优化；以及真实世界执行阶段，使用我们自主实验室中的 Luna qPCR 协议对完整流程进行验证。

语法验证、人工编写的基于规则的验证，以及物理执行过程中的实时视觉观察，而非通过事先仿真来预测并防止错误。因此，当前方法未能利用仿真在执行前验证方面的潜力，仍直接在硬件上执行 AI 生成的协议，尽管这会带来设备损坏、材料浪费和时间成本等风险。

1.4 贡献与目标

当前实验室自动化领域存在一个关键缺口：尚无现有工作将端到端的大语言模型驱动的实验方案生成与基于仿真的错误检测整合到统一流程中，以实现完全自主地生成可信实验方案。数字孪生相关工作能够提供全面的监控和追踪能力，但与自动化实验方案生成系统的集成仍不充分。

PRISM 框架通过一个包含三个阶段的流程弥补了这一差距，该流程整合了多智能体 LLM 规划、由推理模型驱动的实验方案生成，以及基于物理的仿真错误检测。PRISM 将仿真设为执行前的强制关卡，从而建立闭环反馈：检测到的错误将用于指导迭代式协议优化。我们通过聚合酶链式反应扩增（在我们的机器人平台上进行了实验验证）和细胞绘画（经计算验证）展示了该系统的有效性，同时对多种最先进的语言模型进行了比较分析。

这项工作代表着迈向完全自主实验设计与执行的一步，有助于弥合 AI 生成的计划与安全、可靠的实验室自动化之间的鸿沟。

2 方法

我们提出 PRISM：通过智能仿真建模进行协议改进。这是一个能够在由现成的机器人仪器组成的实验室平台上自动完成实验方案设计、验证与执行的框架。我们采用多智能体框架（图 1，第 1 阶段），如第 2.1 节所述，用于生成目标实验流程的步骤；该流程最初以通俗英语描述。这些步骤遵循实验方案生成器能够识别的结构化格式，并可转换为机器人兼容指令。第 2.2 节所述的实验方案生成器（图 1，第 2 阶段）将这些结构化步骤与第 2.3 节所述数字孪生仿真的反馈相结合，通过迭代优化设计，直至生成有效的实验方案。最终生成的有效实验方案随后即可在真实世界实验室中执行（图 1，第 3 阶段），具体如第 2.4 节所述。

2.1 实验方案规划

协议规划阶段侧重于以通俗英语生成完整、结构化的液体处理指令，涵盖

在自动化平台上执行实验所需的全部试剂、实验室器具和操作细节。这些步骤无法直接由机器人执行，但其格式旨在保留机器人化转换所需的动作逻辑顺序和物理约束。

我们评估了两种互补的提示范式：**受限**和**开放式**。完整的提示模板见附录A。在受限设置下，模型会收到明确的试剂体积、孔位布局和步骤格式规则；而开放式提示仅提供高层级的实验目标，因此模型需要推断体积、映射关系和转移序列。我们在两种架构框架——**多智能体系统**和**单智能体系统**——中分别测试每种提示范式，以评估各模型在三个迭代优化周期中生成、优化和自我纠正实验方案的能力。

2.1.1 多智能体与单智能体方案规划框架

我们比较了两种实验方案规划架构：多智能体和单智能体。这两种框架的主要区别在于推理的分配方式，以及错误的检测和纠正方式。

在**多智能体系统**中，任务由四个各司其职的专用智能体分担。*WebSurfer*智能体从在线来源检索实验信息，并提取操作步骤和试剂。随后，*Protocol Planner*智能体将这些非结构化描述转化为结构化计划，选择实验室器具、将试剂分配到实验台上的特定位置，并将每项操作改写为统一的液体处理规范。接着，*Critique*智能体会检查结构化步骤是否存在信息缺失、体积不一致、格式问题和逻辑缺陷。最后，验证代理进行必要的纠正，并将更新后的步骤重新提交给Critique智能体，从而形成优化循环。由于每个智能体仅在其专门职责范围内进行推理，该系统得以发挥模块化优势：推理范围受到约束，错误更易于定位，纠正性反馈也更有针对性。不过，这种设置会引入智能体之间的依赖关系，早期阶段的失误可能会逐级传递，除非在后续检查中被发现。

相比之下，**单智能体**（推理模型）框架依赖单个模型，通过一条连续的思维链完成任务的所有阶段。模型必须一次性检索相关实验细节、推断合适的平台布局、分配试剂、计算体积，并生成一套完整列举且结构化的液体处理步骤，无需外部 Critique 或角色分解。这种方法的优点是简单且执行更快，但也显著增加了模型的认知负荷：所有规划、验证、错误检测和纠正都必须在内部完成。缺少明确的模块边界时，模型可能会忽略不一致之处、忘记先前的约束，或生成逻辑正确但物理上不可行的步骤；随着工作流程变得更长、更复杂，这种情况尤其容易发生。

总体而言，多智能体系统将推理分散到相互协作、彼此强化的专业组件中，而单智能体系统则必须一次性管理整个推理空间。正如我们将在下文详细说明的那样，我们发现，对于依赖关系较少的简短实验方案，例如聚合酶链式反应，单智能体系统的表现相当不错，因为其流程基本固定（制备反应 →热循环），且试剂添加的确切顺序并不重要。相比之下，细胞绘画等工作流程包含许多相互依赖的步骤，其中时序、顺序和试剂相互作用都很重要，因此由单个模型处理时更容易引发连锁推理错误。对于这类复杂且结构化程度更高的实验方案，多智能体框架更加稳健，并能持续生成正确的输出。

2.1.2 评估设置

我们使用五种最先进的大语言模型，在受限提示和开放式提示下对这两种框架进行基准测试：**GPT-5**[21],**ClaudeOpus 4.1**[22], **Claude Sonnet 4.5**[23], **Gemini 2.5Pro**[24],和**Gemini 2.5 Flash**[24]。我们针对每种模型与框架的组合，分别在受限提示和开放式提示配置下进行评估，最多执行三轮优化迭代。如果三轮迭代内未能收敛，则记录最后一次尝试以供分析。对于聚合酶链式反应实验，我们将 GPT-5 的受限多智能体实验方案作为真实标注，因为该方案无需任何纠正循环便完全正确，并经实验生物学家验证。我们将每个生成的实验方案与此参考方案进行比较，依据预先定义的真实标注类别（例如逻辑转移、主混合液计算、试剂和孔位分配、移液器选择及步骤格式），识别正确、缺失和多余的步骤（真正例、假负例和假正例）。随后，我们在这些类别内及类别之间计算 F1分数，以量化生成准确率。

2.2 初始协议生成

为了将英文实验方案转换为 YAML 格式，我们使用了具备推理能力、会在生成响应前进行思考的大语言模型。我们使用六种推理模型测试了框架的这一阶段：**GPT-5**[21], **Claude Opus 4.1**[22], **ClaudeSonnet 4.5**[23], **Gemini 2.5 Pro**[24], **Gemini 2.5 Flash**[24], 和 **Gemini 2.5 Flash-Lite**[25]。

为确保一致性并简化流程，所有模型的生成参数均保持默认值。收到协议规划阶段的输入后，这些模型会将高层次的英文指令转换为结构化的 YAML 输出，具体做法是将指令拆解为可由实验室自动化系统执行的特定机器人命令。生成的 YAML 文件包含按顺序排列的机器人动作及其相关参数，共同构成一份完整的机器可读实验方案。

实验室工作流程十分复杂，因此在许多场景中，仅靠简单的规则将英文转换为机器人命令并不足够。英文中看似简单的指令，例如“封闭微孔板”，往往需要协调执行多个机器人动作，包括从当前位置取出微孔板，将其转移至封板机器人，并执行封板操作。当使用具有特定操作要求的机器人时，复杂性会进一步增加，例如，有些机器人必须先打开才能放入微孔板，且

在执行操作前必须关闭；又或者源机器人和目标机器人各自独特的位置与旋转要求彼此冲突。当实验室中配备了多个具有不同操作约束和空间限制的机器人时，动作顺序就不容易确定，尤其是在机器人之间需要同步时。顺序中只要有一步出错，就可能导致操作在物理上无法实现，例如试图将微孔板放入已关闭的设备，或在机器人未做好准备前就尝试操作。正因如此，对整个工作流程进行推理至关重要：模型不仅必须了解每个机器人的能力，还必须理解它们如何在实验室的空间与操作约束下相互协作。

2.2.1 模型输入

翻译过程使用三类主要输入组件来生成机器人可执行协议。首先，系统接收包含科学参数的英文流程步骤，这些参数包括体积、温度和时间信息，源自前一阶段的实验方案规划。这些步骤描述了需要执行的高层次科学操作，例如本研究所考察的聚合酶链式反应和细胞绘画实验方案。其次，系统会获得有关可用机器人能力的详细信息，包括它们可能执行的动作、每项动作所需的参数以及独特的操作约束。第三，系统接收预期输出格式的规范，其中详细说明了将由实验室自动化系统（阿贡的 MADSci[26] 框架，在早期工作[27]的基础上构建）用于执行实验方案的 YAML文件结构。这些文档可以作为独立文件与 Claude Code 等工具配合使用，从而实现更好的组合与组织；也可以像本研究所采用的方式，在通过直接 API调用模型时，将它们连接成单个提示。

YAML输出格式遵循MADSci框架的实验方案规范，该规范将实验方案编码为按顺序排列的原子机器人操作序列。YAML文件中的每个步骤都包含要使用的机器人、要执行的操作以及该操作所需的任何参数。这种结构化格式将“运行聚合酶链式反应”等高级指令分解为精确的机器人移动、微孔板转移和设备操作序列。YAML结构遵循一致的模式，使实验室自动化系统能够按顺序解析并执行每个步骤，并在步骤之间提供适当的错误处理和报告机会。

生成的 YAML文件由 MADSci 框架负责验证，该框架会进行语法检查和基本能力验证。MADSci 会检查 YAML文件，确保所有必需字段均已填写、实验方案中引用的机器人均存在于实验室配置中，并且请求的操作均在各机器人的能力范围内。这项内置验证省去了在系统中编写单独验证代码的需要。MADSci 还会检查为每项操作提供的参数是否符合相应机器人的预期输入。我们引入的其他更高级验证（例如碰撞检测和物体存在性验证）则在仿真阶段进行，此时会在虚拟环境中对实验方案进行物理执行实验方案在虚拟环境中进行测试。

机器人的信息经过组织，使实验室设置更加灵活且可重复使用。每台机器人的能力、动作、参数和独特的运行要求都会被定义一次并打包在一起。这种方法使研究人员能够将实验室特定限制与机器人的内在属性分别定义。在设置新的实验室配置时，研究人员只需指定所用的机器人，并根据这些预先编写的描述生成提示，然后添加任何实验室特定约束，而无需重新定义机器人的能力。通过重用现有的机器人定义，并仅添加建模特定实验室设置所需的最少信息，这种分离方式使组合新的实验室布局更加容易。

2.3 仿真中的迭代错误检测

为了构建仿真所需的数字孪生环境，我们从供应商或公共资源库获取了机器人的计算机辅助设计（CAD）模型。这些模型最初是静态的，缺少运动仿真所需的物理属性和关节信息。我们通过定义碰撞几何体，并在与真实机器人相匹配的位置设置关节连接，为每个模型添加了必要的物理属性。这一过程需要测量实体机器人的关节位置，并将其映射到CAD模型中的相应位置。我们还开发了仿真驱动程序，将发送给真实机器人的命令转换为适当的控制信号，以驱动 Omniverse[28]，中仿真的关节，使数字机器人能够模仿实体机器人的运动。这些碰撞几何体和关节的补充使仿真能够成为更贴近现实的模型，供后续错误检测步骤使用，同时无需研究人员投入大量额外工作。

搭建实验室场景时，我们测量了现实中的实验室空间，并在仿真中将数字机器人放置在相应位置。CAD模型的高保真度能够实现超越毫米级精度的机器人尺寸和摆放精度，为未来开展更灵活的机器人控制研究提供支持。然而，本次实现侧重于功能准确性，而非精确定位，因为机器人只会按照预定义的配置运动。通过将机器人摆放成与现实视觉上相匹配的布局，我们能够在从机器人中放置和取出微孔板等关键事件发生时可靠地进行碰撞检测，并简化人工目视确认流程仿真是否按预期进行。

2.3.1 仿真到现实的对应关系

MADSci 与仿真环境之间的通信通过定制的机器人接口模块实现，这些模块将 MADSci 默认的物理机器人连接替换为通过 ZeroMQ 向数字孪生发送的定制网络消息。采用这种方式，MADSci 可以像在与真实机器人通信一样运行，但实际上发送的是仿真指令。Omniverse脚本接收这些消息，并指挥仿真的机器人执行请求的动作，随后向 MADSci 发送完成或错误消息。这种通信方式避免了复制每个机器人独特的

网络协议的复杂性，同时利用了经过验证的 MADSci 能力。

我们方法的一个重要特点是，无论控制真实机器人还是仿真机器人，MADSci 的行为都完全一致。我们运行与实体实验室中相同的 MADSci 软件，只需更改机器人接口模块，使其能够与仿真进行通信即可。这意味着，MADSci 内置的所有命令验证、编排逻辑和错误处理，在真实环境中的运行方式与此完全相同。机器人动作的时序与同步均实时进行，消息往返于仿真系统的方式也与实体机器人相同。这种一致性让我们确信，在仿真中正常运行的实验方案，转移到真实实验室后也会有类似表现，因为两者之间唯一的区别只是用仿真机器人替代实体机器人。

2.3.2 错误检测与反馈

仿真通过基于物理属性的碰撞检测来识别多种错误。我们将系统配置为忽略预期碰撞，例如机器人夹爪有意接触其要抓取的微孔板；同时只监控机器人之间，或材料与环境之间的意外碰撞。发生非预期碰撞时，仿真会向 MADSci 报告，并以英文描述问题（指出发生碰撞的对象名称）。此类错误报告会使 MADSci 编排器终止实验执行，就像类似错误发生在实体实验室时一样。系统可以记录直至故障发生时的完整执行日志，既记录错误发生前成功完成的所有步骤，也提供检测到碰撞时正在进行的操作背景。

除了进行碰撞检测外，仿真还会检查物体是否存在，并验证命令是否执行。射线投射系统会在机器人尝试与微孔板或其他材料交互时，检查它们是否存在。如果机器人试图拿起不在预期位置的微孔板，系统便会检测到其缺失并报告错误。同样，如果机器人在当前没有拿着微孔板时尝试放置微孔板，这种不一致也会触发错误。仿真还会根据机器人的物理约束验证命令是否可执行，例如检测请求的位置是否超出机器人的可达范围，或关节限制是否会阻止指定动作。所有错误都会附带清晰易懂的英文描述以及仿真中的相关细节，帮助语言模型更轻松地识别并分析问题。这些运行时检查有助于发现手动检查实验方案时可能不易察觉、但会导致物理执行失败的问题。

2.3.3 迭代过程

仿真检测到错误时，我们会将该错误作为一条新消息添加到与语言模型的对话中，这足以引导模型思考并修复问题。模型在修正实验方案中的错误时会表现出不同的模式，从立即识别并修复问题，到误诊问题并提出无效的解决方案，甚至引入新的错误。借助仿真，通过在每次尝试后反复向模型提示检测到的错误，可以应对所有这些可能的行为模式，让模型探索并从错误中学习。这个过程会一直持续，直到模型收敛到一个没有任何错误的实验方案。

成功的结果表明，协议的物理执行部分应能在真实实验室中顺利运行，因为机器人运动、微孔板转移和设备操作均已在数字孪生中得到验证。不过，需要注意的是，仿真成功并不保证科学准确性。仿真确认了协议在物理层面可执行，但无法验证协议能否达到预期的科学结果，因为我们不会对液体物理学、化学反应或生物过程进行仿真。协议通过仿真验证后，还需经过人工审核，以核实其科学准确性。在我们的工作中，研究人员仍会检查最终协议，确认其按照正确顺序、采用适当参数执行所要求的科学操作。这一人工审核步骤可确保模型在解决物理执行问题的过程中，没有引入会损害实验科学有效性的更改。

2.4 真实世界执行

我们选取了两种具有代表性的实验工作流程来评估 PRISM 的能力：PCR 扩增方案和 Cell Painting检测。我们选择这两种工作流程，是因为它们：(i) 涵盖了多种实验室操作、试剂类型和方案结构；(ii) 广泛应用于生物学研究；(iii) 在结构上各不相同；(iv) 涉及 PRISM 旨在应对的各类推理挑战。PCR 提供了一种紧凑且高度标准化的分子生物学工作流程，而细胞绘画则是一种包含多个步骤和多种试剂、程序复杂度显著更高的成像检测。

我们选择**PCR 扩增** 作为首个基准任务，因为它是一个典型且定义明确的分子生物学程序，具有清晰、可测量的结果。以聚合酶链式反应为基准，可直接与先前的实验方案生成研究进行比较，并为评估机器人系统端到端执行能力提供可复现的基线。在典型的PCR工作流程中，将含有DNA模板、引物、核苷酸和聚合酶的反应混合液配制好，再进行一系列温度循环，以扩增目标DNA。通常通过凝胶电泳进行验证。然而，我们的自动化实验室不具备凝胶处理能力。为实现完全自主执行，我们使用了Luna qPCR Master Mix（New England Biolabs），其中含有嵌入染料，可进行实时荧光读数。这一调整可提供自动化的定量信号，其效果等同于确认扩增产物的存在，且无需任何人工操作。

PRISM生成的实验方案已在我们的自动化实验室平台上端到端执行。该平台集成了OpenTrons OT-2液体处理器[29]，、PF400机械臂[30]，、Hidex酶标仪[31]，、Azenta微孔板封膜仪[32]和撕膜机[33]。这项实验表明，PRISM能够生成机器人可执行协议，并使其在无需人工干预的情况下运行至完成。

为评估 PRISM 在聚合酶链式反应之外的通用性，我们还使用 PRISM 来

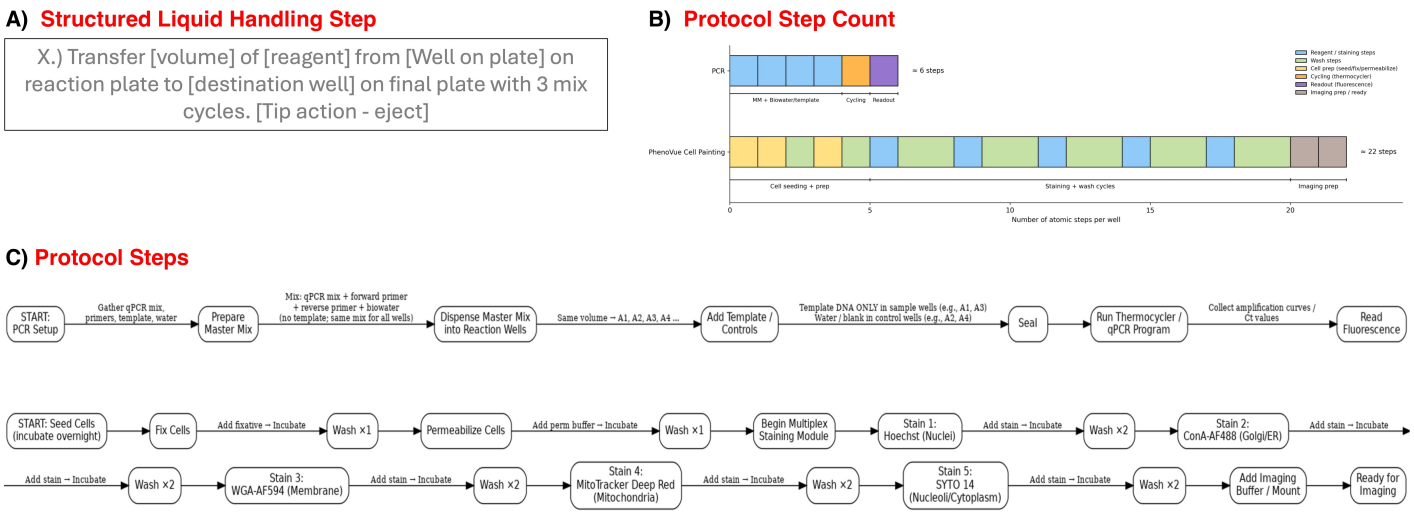


图 2 Luna qPCR 与 PhenoVue 细胞绘画实验的实验方案复杂度比较。(A) 所有生成的实验方案都必须遵循的结构化液体处理步骤格式。(B) 各项检测每孔/每反应所需的原子步骤数量，按操作类型着色。与 qPCR 相比，细胞绘画需要添加更多试剂并进行更多洗涤循环，而且对顺序有严格要求（例如，染色–洗涤循环必须依次进行）；而在 qPCR 中，试剂添加顺序无关紧要。(C) qPCR（上）和细胞绘画（下）的分步工作流程示意图，凸显了细胞绘画实验方案在长度、重复次数和顺序约束方面的增加。

为 **Cell Painting检测**生成实验方案。Cell Painting检测是一种标准的形态分析工作流程，广泛用于高内涵成像筛查。在典型的Cell Painting检测中（见图 2），先接种贴壁细胞，然后固定、通透化，并使用一组荧光染料进行染色，以突出细胞核、内质网、高尔基体和线粒体等细胞结构。随后，使用高内涵筛查显微镜对染色后的细胞进行成像，以提取形态特征。Cell Painting考验了聚合酶链式反应不会涉及的实验方案生成能力：多步洗涤和孵育、依赖顺序的操作、复杂的试剂处理，以及冗长且包含分支的工作流程。这些特点使其成为检验 PRISM 处理流程复杂、多模块实验方案能力的理想测试。

我们通过计算机模拟评估了 PRISM 生成的细胞绘画实验方案。领域专家审查了生成步骤的生物学正确性和内部一致性。尽管我们可以通过受控移液，在 OT-2 上模拟细胞绘画的核心步骤（固定、透化、染色和洗涤），但目前的工作单元并未配备集成式自动洗板机。由于细胞绘画通常涉及许多温和的洗涤循环，以尽可能保留贴壁细胞单层，因此高内涵筛选工作流程中通常会使用专用洗板机。出于这一原因，加之我们的平台缺少高内涵成像系统，我们通过计算机模拟而非实际操作来验证细胞绘画实验方案。

3 结果与讨论

我们按照 PRISM 框架三阶段流程（如图 1 所示）的顺序呈现结果。第 3.1 节评估第 1 阶段，即考察不同的实验方案规划方法，这些方法将高层次的实验意图转化为结构化的英文指令。我们还比较了多智能体与单智能体框架，以及受约束与开放式提示策略。第 3.2 节对应第 2 阶段，即实验方案生成与验证循环：将结构化步骤转换为机器人可执行的 YAML 格式，并通过基于仿真的错误检测反复优化。我们还展示了一项消融研究，说明仿真对于实现物理正确性不可或缺。最后，第 3.3 节介绍第 3 阶段：在自动化实验室平台上执行经过仿真验证的实验方案，以展示端到端功能和生物学有效性。

3.1 实验方案规划

我们首先评估 PRISM 框架的初始阶段：将科学意图转化为结构化实验计划。此阶段构成基础逻辑层；在此处引入的错误会向下游传播，使得即使语法完全正确的机器人代码在科学上也可能无效。为评估这一能力，我们从两个维度对性能进行基准测试：架构框架（比较单智能体推理与所提出的多智能体系统）以及提示约束（评估在严格指令和开放式指令下的适应能力）。

3.1.1 实验设置与评估标准

我们评估了五种大语言模型（GPT-5、Claude Opus 4.1、Claude Sonnet 4.5、Gemini 2.5 Pro 和 Gemini 2.5 Flash）能否以通俗英语，为第 2.4 节所述的聚合酶链式反应和细胞绘画工作流程生成结构化液体处理步骤。

对于聚合酶链式反应，每个模型均在多智能体和单智能体框架下，分别采用受限和开放式提示范式进行测试。性能根据以下两项进行评估：（1）完善生成的实验方案所需的修正迭代次数；（2）生成步骤中存在的逻辑错误、格式错误或物理错误类型。我们最多允许进行三次修正迭代；如果模型在此限度内未能生成有效的实验方案，则将该次尝试记录为未

Ground Truth Criteria			GPT 5	Claude Opus 4.1	Claude Sonnet 4.5	Gemini 2.5 Pro	Gemini 2.5 Flash
Multi-Agent workflow	Constrained	Well Assignment	✓	✓	✓	✓	✓
		Reagent Volumes	✓	✓	✓	✓	×
		Exactly 3 Thermocycling Steps	✓	×	×	×	✓
		Step Format	✓	×	×	×	×
		Reagent Logic	✓	✓	✓	✓	×
		Practical Pipette Use	✓	✓	✓	✓	✓
		F1 Score	**1.0**	**0.80**	**0.80**	**0.80**	**0.67**
	Open -Ended	Practical Well Assignment	✓	✓	✓	✓	✓
		Adequate Well Spacing	✓	✓	✓	✓	×
		Correct Volume Calculations	✓	×	×	×	×
		Exactly 3 Thermocycling Steps	✓	×	×	×	×
		Step Format	✓	×	×	×	×
		Reagent Logic	✓	✓	✓	×	×
		Practical Pipette Use	✓	✓	✓	×	×
		Volume and capacity compliance	✓	✓	✓	×	×
		F1 Score	**1.0**	**0.77**	**0.77**	**0.40**	**0.22**

(a) 多智能体

Ground Truth Criteria			GPT 5	Claude Opus 4.1	Claude Sonnet 4.5	Gemini 2.5 Pro	Gemini 2.5 Flash
Single Agent Workflow	Constrained	Well Assignment	✓	✓	✓	✓	✓
		Reagent Volumes	✓	✓	✓	✓	✓
		Exactly 3 Thermocycling Steps	✓	✓	×	✓	×
		Step Format	✓	×	×	×	×
		Reagent Logic	✓	✓	✓	✓	✓
		Pipette Use	✓	✓	✓	✓	✓
		F1 Score	**1.00**	**0.91**	**0.80**	**0.91**	**0.80**
	Open -Ended	Practical Well Assignment	✓	✓	✓	✓	✓
		Adequate Well Spacing	✓	✓	✓	✓	✓
		Correct Volume Calculations	×	×	×	✓	✓
		Exactly 3 Thermocycling Steps	×	✓	×	×	×
		Step Format	✓	×	×	×	×
		Reagent Logic	×	×	×	×	×
		Pipette Use	×	×	×	×	×
		Volume and capacity compliance	✓	✓	✓	✓	✓
		F1 Score	**0.67**	**0.67**	**0.55**	**0.67**	**0.67**

(b) 单智能体

图 3 多智能体与单智能体工作流程在 Luna qPCR 协议生成中的性能比较。每个面板总结了五种大语言模型（GPT-5、Claude Opus 4.1、Claude Sonnet 4.5、Gemini 2.5 Pro 和 Gemini 2.5 Flash）在所有真实标注评估标准上的模型准确率。✓ 和 表示是否满足或违反了每项标准。颜色编码表示相对于真实标注的偏差类型：黄色 =额外操作（假阳性），红色 = 缺失操作（假阴性）。这些错误计入底行报告的 F1分数，该分数用于量化整体实验方案准确性。面板 (a) 展示了多智能体工作流程（WebSurfer → Planner → Critique → Validator），面板 (b) 展示了未对任务进行模块化分解的单智能体工作流程。比较结果表明，与单体式推理生成相比，结构化的多智能体推理能实现更高的正确率和更可靠的协议合成。

收敛。聚合酶链式反应的结果由一位生物学家审核，并在我们的机器人平台上进行了实验验证。

我们评估了每个生成的实验方案的结构正确性和生物学可行性。在**受限提示**设置下，模型接收固定的实验参数、试剂映射和孔位布局，以尽量减少歧义，确保成功反映的是对给定指令的精准遵循，而非创造性推断。我们根据六项预定义标准评估表现：（1）源孔和目标孔分配正确；（2）试剂体积正确；（3）恰好包含三个热循环步骤；（4）遵循规定的步骤格式；（5）对测试孔和对照孔使用正确的试剂；（6）移液器选择恰当。

相比之下，**开放式提示**设置仅提供高层次的实验目标，而不固定试剂映射或各组分的体积，同时仍明确规定了关键实验室约束，例如 OT-2 操作台布局（如反应板和目标板的槽位分配）以及目标总反应体积。因此，模型需要推断出兼容的体积分配、孔位布局和转移序列，以满足这些物理和生物学约束。此设置下的评估侧重于八项能够体现推理灵活性和实际可行性的标准：（1）实用的孔位映射；（2）测试孔与对照孔之间保持足够间距以避免交叉污染；（3）符合生物学要求的体积计算；（4）恰好包含三个热循环步骤；（5）正确的步骤格式；（6）测试孔与对照孔的试剂逻辑正确；（7）正确且一致地使用移液器；以及（8）符合最小体积和板容量限制。

附录 A 中提供了受限和开放式范式所用的完整提示模板和模型指令。对于上述各项标准，我们使用 F1分数衡量定量准确性；该分数同时反映生成的实验方案相对于

表 1 Luna qPCRProtocol结果：在受限和开放式提示下，各模型分别在多智能体和单智能体框架中实现收敛所需的细化迭代次数（1–3）。 ✓ 表示模型首次尝试即生成了正确的实验方案，无需细化；整数（1–3）表示收敛前所需的细化循环次数；而

† 表示在三次迭代内未能收敛。

	多智能体		单智能体	
模型	受限	开放式	受约束	开放式
GPT-5	✓	✓	✓	2
Claude Opus 4.1	2	†	†	†
Claude Sonnet 4.5	†	†	†	†
Gemini 2.5 Pro	†	†	†	†
Gemini 2.5 Flash	†	†	†	†

真实参考答案的精确率和完整性。F1分数的计算方式如下：

$$F_1 = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

其中，TP、FP 和 FN 分别表示正确步骤、多余步骤和缺失步骤的数量。F1分数综合考虑了各提示范式所选真实标准中的新增、删除和错误。

3.1.2 用例 1：直观探索 PRISM 的能力

我们首先在快速原型实验室 (RPL) 中，使用一组孔板图案任务评估 PRISM，以展示其在生物实验室之外生成正确 OT-2协议的能力。具体而言，我们通过复现 Boiko等人此前展示的几何图案任务，验证了 PRISM 的空间规划能力[34], ，证实我们的系统能够成功将高层次的视觉描述转化为机器人可执行的实验方案。这些非生物学演示表明，仅凭自然语言目标，PRISM 就能推断孔位、计算体积、指定试剂来源，并输出机器人可执行的液体处理指令。图 4 展示了所用的四种图案：(A) 左上角的 3 × 3彩色方块，(B) 从微孔板左下角开始的对角线，(C) 隔行填充图案，以及 (D) 通过对每隔一个孔应用颜色渐变而形成的彩虹棋盘格。这些任务凸显了 PRISM 超越检测特定工作流程进行泛化、并为任意用户定义目标生成实验方案的能力。只需用英文描述要填充哪些孔以及如何为每个孔着色，PRISM 就能确定精确的孔位和输入液体的比例，从而完成任务。

3.1.3 用例 2：PCR协议生成

接下来，我们在 RAPID-446（我们的生物安全一级实验室）中评估了 PRISM，以展示 PRISM 能够推广到不同的实验室配置和不同的仪器。RAPID-446 配备了与 RPL 不同物理布局中的额外机器人，使我们能够证明 PRISM 并不局限于我们在开发过程中使用的 RPL 配置。

图 3a 和图 3b 汇总了两种框架下各模型在聚合酶链式反应中的 F1 准确率，而表 1 列出了每个模型收敛所需的修正迭代次数。这些分析共同反映了实验方案生成的准确性，以及各框架在迭代优化过程中的适应能力。

在受限提示下，GPT-5 在多智能体和单智能体框架中均获得了完美的 F1分数，并在首次尝试时生成了正确的实验方案。在开放式提示下，GPT-5 最初在体积计算、步骤格式、热循环步骤数量和试剂转移方面表现出轻微不一致，但在三次优化迭代内收敛为完全正确的实验方案。

相比之下，Claude 和 Gemini 模型在多次迭代中未能可靠收敛，若干配置的性能还会随着时间推移而下降。例如，Gemini 2.5 Flash（使用受限提示进行推理）、Claude Opus 4.1（使用固定约束进行推理）以及 Claude Opus 4.1（开放式多智能体框架）经常添加冗余步骤，将体积放大到不切实际的水平，或修改此前正确的操作。这些模型还难以维持所要求的步骤格式，并且在受限提示下无法生成恰好三个热循环步骤。当使用开放式提示时，它们在各项评估标准上的偏离进一步加剧，尤其是在维持与机器人平台兼容且符合生物学要求的体积或浓度计算方面。总体而言，多智能体框架仍然为大多数非 GPT 模型实现了更高的准确率和更快的收敛，这主要是因为其结构化反馈能够帮助模型纠正格式和逻辑不一致之处，而单一推理链无法自行纠正这些问题。然而，它们在步骤格式方面仍存在轻微问题，因此我们在表 1 中称其尚未完全收敛。

这些结果表明，模型性能不仅受参数容量影响，也取决于推理动态：GPT 模型展现出持续的上下文推理能力，使其能够隐式处理实验约束；而 Gemini 和 Claude 等倾向于更快生成响应的模型，往往会丢失上下文，或在内部 deliberation 不充分的情况下作答，因此需要更明确的结构和纠正性反馈，才能生成有效协议。

3.1.4 用例 3：细胞绘画

为评估在更复杂实验工作流程中的表现，我们采用多智能体和单智能体框架生成了细胞绘画实验方案。此次评估未针对不同提示范式进行基准测试，因为目标是检验各框架能否处理更长、包含多个步骤的流程，而非优化特定的提示条件。生物学家审查了生成的实验方案，以确认其在生物学上合理，且在我们的自动化平台上物理可行。在这种情况下，随着操作数量增加，推理模型往往无法保持连贯的步骤结构，经常遗漏试剂或错置操作步骤。相比之下，多智能体框架生成的实验方案始终一致、逻辑清晰，并涵盖了必要的试剂添加和清洗步骤顺序。将任务分配给专门的智能体，有助于在操作复杂度提高时保持结构和正确性，凸显了模块化、反馈驱动的设计对于规划更大型、多阶段实验的重要性。

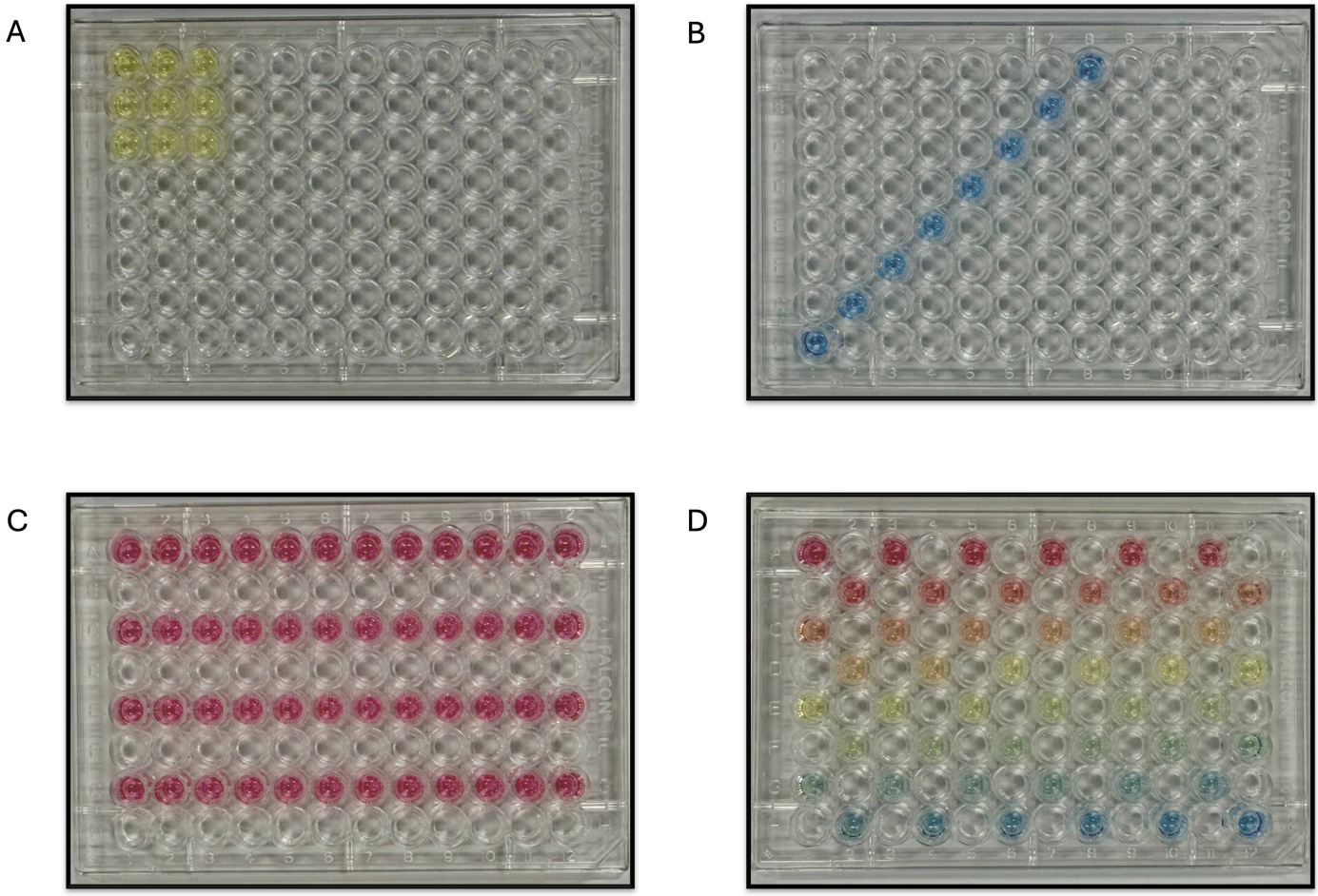


图 4 用于评估 PRISM 的可视化模式生成任务。（A）左上角的一个 3 × 3 彩色方块。（B）从左下角（H1）开始的对角线模式。（C）交替行填充模式。（D）彩虹棋盘格，将红–黄–蓝渐变与基于奇偶性的填充规则相结合。

总的来说，这些结果表明，结构化提示和模块化任务分解在生成正确且可实际执行的实验方案方面发挥着互补作用。GPT-5等高容量模型在两种框架和不同提示类型下都表现可靠，而较小的模型则能从多智能体系统提供的结构化反馈和受约束提示的明确引导中获益良多。当实验方案变得更长或更详细时，多智能体框架优势明显，能够持续纠正推理模型单独使用时难以解决的逻辑错误和格式错误。综合来看，这些发现证明了实验方案规划准确且适应性强，并为下一阶段将结构化步骤转化为完全可执行的机器人工作流奠定了基础。

3.2 实验方案生成/验证

为评估各模型的初始协议生成能力，我们考察了它们能否根据结构化英文指令，一次性生成完整且可执行的 PCR 工作流程。这项零样本评估为衡量各模型对机器人约束、测序要求以及实验室环境物理限制的理解提供了基准。

图 5 对六种受测推理模型各自首次生成的实验方案与真实工作流进行了比较分析。

分析揭示了各模型之间一个显著的模式：六个受测系统中有五个（Claude Opus 4.1、Claude Sonnet 4.5、Gemini 2.5 Pro、Gemini 2.5 Flash 和 Gemini 2.5 Flash-Lite）都遗漏了在将微孔板放入热循环仪和酶标仪之前所需的打开指令。这类错误代表着一种物理上的不可能：尝试将微孔板插入关闭的设备会导致实验失败，并可能损坏实体实验室中的设备。该错误模式表明，模型在长时程规划方面存在一致性缺陷：尽管模型成功生成了大部分基本的协议结构，但未能预见某些设备在交互前需要额外操作这一要求，而这一约束不同于其他实验室机器人的典型使用模式。

值得注意的是，GPT-5 在初始生成阶段取得了完美表现，F1分数为 1.0，首次尝试便生成了一个无需任何错误即可执行的物理实验方案。不同模型之间的性能梯度体现在 F1分数从 GPT-5（1.0）经 Claude Opus 4.1（0.94）下降至 Gemini 2.5 Flash-Lite（0.82），定量展示了模型能力如何直接影响初始

Ground Truth	GPT-5	Opus 4.1	Sonnet 4.5	Gemini 2.5 Pro	Gemini 2.5 Flash	Gemini 2.5 Flash-Lite
1. ot2 run_protocol	✔	✔	✔	✔	✔	✔
2. transfer → exchange	✔	✔	✔	✔	✔	✔
3. transfer → sealer	✔	✔	✔	✔	✔	✔
4. seal	✔	✔	✔	✔	✔	✔
5. open thermocycler	✔					
6. transfer → exchange	✔	✔	✔	✔	✔	✔
7. transfer → thermocycler	✔	✔	✔	✔	✔	✔
			5	5	5	5
8. close thermocycler	✔	✔	✔	✔	✔	✔
9. run thermocycler	✔	✔	✔	✔	✔	✔
10. open thermocycler	✔	✔	✔	✔	✔	✔
11. transfer → exchange	✔	✔	✔	✔	✔	✔
12. transfer → peeler	✔	✔	✔	✔	✔	
13. peel	✔	✔	✔	✔	✔	
	+transfer		+transfer			
14. open hidex	✔					
15. transfer → hidex	✔	✔	✔	✔	✔	✔
			14	14	14	14
16. close hidex	✔	✔	✔	✔	✔	✔
17. run hidex	✔	✔	✔	✔	✔	✔
18. open hidex	✔	✔	✔	✔	✔	✔
F1 Score	1.00	0.94	0.88	0.88	0.88	0.82

Missing action (False Negative) Extra action (False Positive) Benign insertion

图5 PCR工作流程的初始生成实验方案与真实标注对比。每一列展示了指定推理模型生成的第一个实验方案，并与正确的动作序列进行对比。 ✓ 表示操作位置正确。红色单元格表示缺少操作，黄色单元格表示插入了操作（数字对应操作的正确位置），蓝色单元格表示无害修改，即模型添加了不必要但无害的中间步骤。GPT-5 实现了完美的初始生成，而其他模型则表现出一种常见的失败模式：在将微孔板转移到热循环仪（第5步）和酶标仪（第14步）之前，遗漏了所需的打开指令。

生成质量。能力较弱的模型出现遗漏动作的比率更高，也未能完全捕捉基本的协议结构。

这些零样本结果揭示了 PRISM 框架所要解决的根本挑战：即使是最先进的语言模型，在将高层次的科学意图转化为可执行的机器人指令时，也经常会生成物理上不可行的实验方案。如果直接在物理硬件上执行，图 5 所示的错误将导致执行失败，这也凸显了 PRISM 基于仿真的迭代优化循环的核心作用——在真实世界执行之前，迭代检测并纠正这些缺陷。

3.2.1 通过仿真反馈进行迭代优化

除 GPT-5 外，所有模型都需要在仿真反馈的帮助下进一步迭代，才能生成有效实验方案。仿真环境检测出物理上不可能的情况，例如试图将微孔板放入已关闭的设备中，并返回描述具体碰撞或约束违反的错误消息。模型展现出了解读这些错误消息、识别出问题的步骤并提出修正方案的能力，不过错误纠正的效率和模式因模型而异。

表 2 各模型在仿真中生成物理属性有效的 PCR实验方案所需的仿真迭代次数。GPT-5 在首次仿真运行中便生成了正确的实验方案，没有出现任何错误；而较弱的模型则需要逐渐增加尝试次数，才能找出并纠正所有错误。Gemini 2.5 Flash-Lite 在 3 次迭代中通过了所有仿真检查，但未包含剥离命令。该命令是确保酶标仪正确读取结果所必需的科学步骤，而我们的物理仿真并未对其进行验证。

模型	总迭代次数
GPT-5	1
Claude Opus 4.1	2
Gemini 2.5 Pro	2
Claude Sonnet 4.5	3
Gemini 2.5 Flash	3
Gemini 2.5 Flash-Lite	3（未满足科学要求）

中等能力的模型 Claude Sonnet 4.5 和 Gemini 2.5 Flash 成功解读了模拟错误消息，并正确诊断出失败的根本原因。然而，这些模型表现出一种局部修正模式：当仿真指出某个错误的具体实例时（例如，尝试在特定步骤将微孔板放入已关闭的设备中，如图 6 所示），模型只会修复该特定实例。随后进行的仿真运行会针对其他出现相同错误的实例生成新的错误消息，因此需要额外迭代，以系统地消除该错误模式的所有出现位置。这种行为导致 Claude Sonnet 4.5 和 Gemini 2.5 Flash 均需要经过三次迭代才能获得有效协议，如表 2 所示。

相比之下，能力更强的模型 Claude Opus 4.1 和 Gemini 2.5 Pro 展现出了全局模式识别能力。当收到关于某一处错误实例的错误消息时，这些模型能够识别出它们在实验方案的多个位置犯下了同样的错误，并在一次迭代中同时修正所有实例。这种从一个错误实例进行泛化、识别并修复系统性问题模式的能力，使这些模型仅经过两次迭代就实现了有效协议。

在测试的模型中，能力最弱的 Gemini 2.5 Flash-Lite 在错误诊断方面的表现明显较差。它没有立即识别出正确的根本原因，而是错误地处理了初始错误，并在此过程中引入了新的错误。直到从后续错误消息中积累了更多上下文后，该模型才找到一个物理上可行的解决方案。这种试错方法经过三次迭代，才解决仿真检测出的基于物理的错误。不过，尽管该模型生成的实验方案在物理上可执行，并通过了所有仿真检查，但仍未包含剥离命令。对于酶标仪得出正确结果而言，这一步在科学上必不可少，而我们的物理仿真并未对此进行验证。当人类研究人员提供了一条手动格式化的错误消息，指出缺少这项操作后，Gemini 2.5 Flash-Lite 成功在最后一次迭代中进行了修正。这一完整的完善过

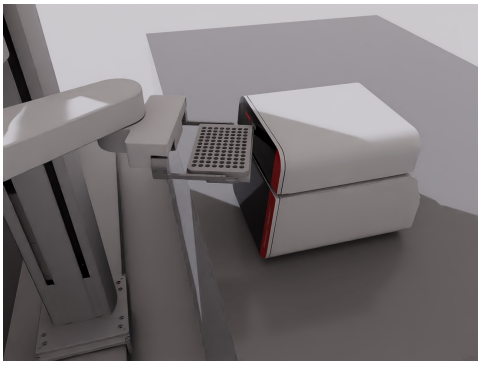


图6 仿真环境中的板碰撞错误示意图。设备盖仍处于关闭状态时，机械臂尝试将微孔板插入热循环仪，这在物理上是不可能的。当模型未能在插入微孔板前加入必需的打开指令时，就会发生此错误。

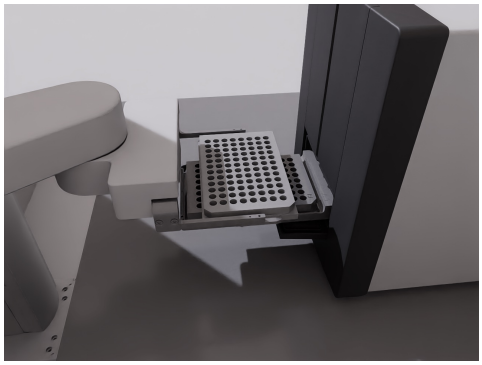


图7 仿真环境中板方向不匹配错误的可视化。转移操作期间，微孔板的方向不正确，导致一种在执行过程中会与机械臂或设备发生碰撞的空间配置。尽管经过多轮自我纠错，模型仍宣称实验方案已准备好执行，但这类细微的几何约束违反始终未被检测到，展现了纯文本推理在验证物理可行性方面的局限性。

程总共经过四次迭代，最终得到了一份在物理上有效且科学上完整的实验方案。这表明，虽然基于仿真的反馈能显著帮助较弱的模型实现物理正确性，但仍不能免除对最终实验方案进行科学审查的必要。

为评估这些结果在聚合酶链式反应实验之外的普适性，我们还使用 Cell Painting协议评估了所有模型。在此情形下，每个模型都首次尝试便生成了正确的实验方案，验证无需迭代。这种普遍成功是因为 Cell Painting协议的大部分复杂性都在于液体处理步骤，而这些步骤已在规划阶段得到充分覆盖（图1，阶段1），如图2所示；此外，Cell Painting工作流程中的机器人种类较少，不像聚合酶链式反应实验方案那样，机器人之间需要复杂协调，而这正是导致模型出现大多数错误的原因。尽管所有模型首次尝试均获成功，模型行为仍存在一些细微差异。各模型独立选择了对室温的理解，不同模型给出的数值在22至25摄氏度之间。此外，当被要求将微孔板在4摄氏度下封板并无限期储存时，除 GPT-5 外的所有模型都选择封板后不将其放入培养箱，而是留下注释，提示人类研究人员按需设置储存条件。这种做法体现了对输入文档的保守解读，因为文档并未说明无限时长孵育是受支持的功能。GPT-5 则选择将密封微孔板放入培养箱，在要求的温度下存放指定时长（24小时），并附注提示研究人员根据需要调整持续时间。

3.2.2 无仿真的自我纠错的局限性

为评估基于仿真的验证相较于语言模型自我批评所带来的额外价值，我们开展了一项消融研究，其中禁用了仿真环境。该实验检验了这样一种可能性：推理模型在收到提示、需要识别并修正自身错误后，能否达到与包含仿真反馈的完整 PRISM 框架相同的物理正确性水平。在这项工作中，我们设计了一个更具挑战性的场景，以压力测试模型的抽象推理能力：一个依次通过 PCR 工作流程处理两个微孔板的实验方案，由此产生冲突——两个微孔板需要同时占用酶标仪，这在物理上是不可能的。该场景模拟了一种现实的失效模式：上游实验方案规划阶段生成的方案在逻辑上连贯，却无法在物理上实现。

我们向 Claude Sonnet 4.5 提供了这一存在缺陷的计划，并要求其生成相应的 YAML 协议。初始生成结果包含多个错误：模型忘记在尝试插入微孔板之前，为热循环仪和酶标仪分别加入打开指令；同时，它生成了不匹配的微孔板转移方向，这将导致碰撞。此外，该实验方案的终止状态显示两块微孔板同时位于酶标仪中，证实模型在生成过程中未能检测到这一物理上的不可能情况。

当被要求检查自己的工作并找出错误时，模型成功识别出一个高层次的逻辑冲突：两块微孔板无法占据同一位置。它提出先将第一块微孔板移至交换台，再处理第二块微孔板，展现出对空间冲突进行逻辑推理的能力。然而，这一修改又引入了新的问题。实验方案要求将交换台用作转移第二块微孔板时的中间位置，而第一块微孔板此时占据了该位置，导致交换台被占用。在第二次收到自我批判请求后，模型意识到了这一新出现的冲突，并再次修订计划，安排将第一块微孔板移回 OT-2 中的原始位置，从而腾出酶标仪和交换台。

经过这一系列修正后，模型进行了最终的自我审查，并宣布该实验方案正确且已准备好执行。然而，多个物理上不可能的情况仍未被发现。该实验方案仍试图将微孔板放入已关闭的设备中，这与图 5 所示的错误模式相同：除 GPT-5 外，所有模型在初始生成中都出现了这种错误。此

外，微孔板转移方向不匹配的问题仍然存在（图 7），这将在执行过程中导致碰撞故障。

在完整的 PRISM 框架中，这个被模型判定为“正确”的实验方案，在仿真环境中执行时会失败。基于物理属性的接地错误消息提供了一种与自我批判截然不同的验证方式，使模型能够识别并纠正那些仅靠工具辅助之外的推理无法发现的错误。

这项消融研究揭示了语言模型自我纠正用于协议优化时的局限性。模型展现出了一定的复杂逻辑推理能力，通过多轮抽象问题求解，成功识别并解决了高层次的空间冲突。然而，鉴于多步骤协调任务的复杂性，以及某些物理约束的微妙特性，仅凭基于文本的推理不足以实现物理上可执行的结果。

这一结果表明，PRISM 的仿真组件提供了一个有价值的验证层。虽然语言模型具备出色的推理能力，并能通过自我批判识别多种错误，但任务的复杂性与验证物理可行性的需求相结合，意味着必须依托基于物理属性的环境。仿真是对模型推理的补充，而非替代；它能够提供物理验证，而对于复杂的多步骤机器人实验流程，仅靠工具辅助之外的推理无法可靠地实现这一点。

3.3 真实世界实验验证

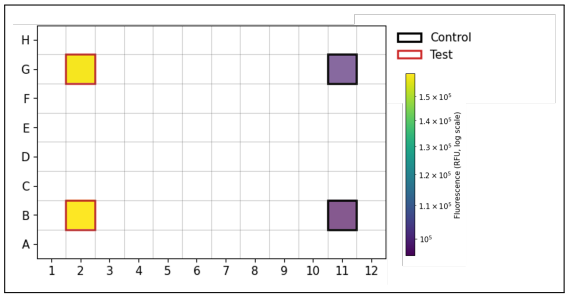


图8 PRISM执行的实验方案所得PCR板荧光读数。测试孔（红色轮廓）显示出扩增信号，而对照孔（黑色轮廓）则保持在基线水平。荧光以对数刻度的相对荧光单位（RFU）表示。

我们在自动化实验室平台上执行了PRISM生成的PCR实验方案，如第3.1.3节所述。我们使用了Luna qPCR主混合液，它支持以基于荧光的验证取代凝胶电泳，从而实现全自动执行。如图8所示，测试孔（红色轮廓）产生了强烈的扩增信号，而对照孔（黑色轮廓）则保持在基线荧光水平。测得的RFU值与科学家手动进行相同PCR检测时所得结果一致，这表明PRISM生成的实验方案能够产生具有生物学效度的结果，与传统手动实验相当。

为评估 PRISM 在聚合酶链式反应之外的通用性，我们使用该框架生成了细胞绘画检测实验方案，如第3.1.4节所述。PRISM 成功生成了一套详细的机器人指令，涵盖试剂处理、微孔板设置和成像准备。这些指令经过计算机模拟验证，并由领域专家审核。

总而言之，这些结果证实，PRISM能够在自动化实验室中提供端到端可执行的实验方案，并且还能从单步分子检测推广到更复杂的多步骤生物工作流程。

4 结论

我们提出了 PRISM，这是一个通过将基于大语言模型的规划系统与基于仿真的验证相结合，解决如何将科学假设转化为无错误机器人指令这一挑战的框架。这种集成确保实验方案经过物理可靠性审查，使生成式人工智能能够应用于实验室环境，否则可靠性方面的顾虑会阻碍其应用。以下将总结系统在各项关键基准测试中的表现，综合关于智能体推理所获得的见解，并概述实现完全自驱动实验室仍面临的挑战。

4.1 系统性能与贡献

在本研究中，我们展示了大语言模型成功应用于实验室研究的自动化实验方案生成，并建立了一个弥合高层实验意图与可执行机器人指令之间巨大差距的框架。PRISM 借助 MADSci 平台，展现出与现有自动化实验室硬件的兼容性，这表明该方法无需专用或定制的机器人系统也能有效运行。通过集成基于 NVIDIA Omniverse 和 Isaac Sim 的仿真优化，我们验证了一种可在物理执行前迭代改进实验方案的方法，从而降低设备损坏和实验失败的风险。

我们还报告了多种大语言模型之间的定量性能比较结果。这项研究揭示了不同推理模型在实验室自动化方面各自独特的能力与局限性。该系统的有效性通过在我们的自动化实验室中成功执行聚合酶链式反应，明确地与现实世界验证联系起来；同时，其通用性则通过在计算机模拟中生成更复杂的 Cell Painting协议得到了证明。

4.2 关键发现与见解

协议规划评估揭示了语言模型在将实验程序分解为可执行步骤时所采用方法的根本差异。多智能体工作流通过强制实行明确的关注点分离，展现出更优异的性能：检索程序性知识；对平台布局和实验室器皿定位进行空间推理；在物理约束下进行逐步格式化；并系统地验证完整性和逻辑流程。与单智能体推理相比，这种结构化方法减少了假阳性（不必要的操作）和假阴性（遗漏关键步骤）。单智能体推理往往会生成过于冗长或缺少必要细节的实验方案。值得注意的是，迭代细化

机制——由批评与验证智能体向 Planner 提供反馈——被证明对于实现收敛至关重要，大多数模型需要经过 2 或 3 个细化循环，才能达到可接受的准确度水平。对受约束提示与开放式提示的比较表明，受约束提示通过提供明确指导提高了初始生成质量，而开放式提示则检验了模型独立推理实验要求、体积和转移序列的能力。这一发现表明，协议规划性能不仅取决于模型能力，还取决于通过提示设计和智能体专业化所提供的结构化支架程度。

我们的评估揭示了不同模型在实验方案生成各个方面的特定优势，其初始生成质量和错误纠正能力也各不相同。所有受测模型都证明，迭代优化有助于生成有效实验方案，但每轮迭代的提升幅度因模型推理能力而显著不同。较弱的模型需要多轮迭代才能纠正同一错误的各个实例，而较强的模型则能同时识别并修复所有实例。正如消融研究所示，移除仿真反馈后，中等能力模型的表现降至借助仿真的低能力模型水平。仿真对于防止现实世界中的失败至关重要，它能够可靠地检测物理执行中的问题，例如碰撞、未能为具有特定操作约束的机器人添加打开步骤，以及可能导致实体实验室中操作失败的无效指令序列。重要的是，PCR实验的表现与人工操作相当，这表明通过 LLM 驱动的实验方案生成实现自动化并不会损害生物学结果，而且能够取得与传统人工方法相同的科学结果。

4.3 当前局限与未来工作

尽管取得了令人鼓舞的成果，仍有若干局限值得在未来工作中加以关注。大语言模型仍需经过大量提示工程，才能实现稳定可靠的性能，而且不同模型系列对提示措辞的敏感程度差异显著。针对实验室协议开发自适应提示策略或领域专用微调方法，或可减少对人工提示设计的依赖。目前的模型在推理多孔操作、复杂体积计算以及多对多转移模式方面仍面临困难，这限制了它们向更复杂分析测定泛化的能力。要将自动化实验方案生成扩展至日益复杂的实验工作流程，解决这些挑战至关重要。

液体世界模型的开发代表了未来研究的另一个有前景方向。这类模型能够捕捉当前模拟中尚未涵盖的物理现象，包括黏度效应、蒸发速率和污染风险，从而在物理执行实验方案之前实现更全面的验证。除基于文本的输入之外，纳入实验装置图像等其他模态，实验室设备的CAD模型以及实时传感器数据，可能会提高实验方案生成的稳健性和准确性。

当前 Omniverse 仿真存在范围局限，尤其是在通过可达性和碰撞反馈进行物理运动验证方面。仿真保真度也受到一些限制，例如缺乏液体物理建模，且无法仿真化学或生物过程，因此科学准确性必须通过其他方式验证。扩展 Omniverse 的错误检测能力，使其涵盖科学准确性验证和更详细的实验方案遵循情况验证，将增强系统在实际执行前发现错误的能力。

我们还注意到，对于那些未能在文本输出格式中清晰表达的错误，当前基于文本的仿真反馈限制了人们对复杂实验方案执行流程的直观理解。Agentic Lab[15] 等近期研究展示了视觉语言模型在实验室推理方面的潜力，这为 PRISM 的未来发展指明了方向：纳入多模态仿真反馈——在文本说明之外，将预测的设备状态、时序关系和潜在失效模式可视化，以促进实验方案优化。

最后，我们观察到，此处考察的仿真环境还可用于优化实验室设计；借助数字孪生建模，可以帮助识别能够最大限度提高效率，并尽量减少机器人之间潜在碰撞或通行冲突的实验室布局。

4.4 更广泛的影响

PRISM 框架有望通过降低缺乏机器人学和自动化编程专门知识的机构与研究人员的入门门槛，推动实验室研究自动化的普及。PRISM 方法使科学家能够用结构化自然语言描述实验，而不必掌握机器人控制系统的详细知识，从而让更广泛的群体能够使用实验室自动化。这种普及对于生物实验方案的可重复性和标准化具有重要意义，因为自动生成的实验方案可以在不同实验室之间共享、验证并一致地执行，前提是这些实验室配备兼容的自动化平台。

本文提出了一条通往完全自主实验设计与执行的路径，使从假设形成到实验执行和数据分析的整个流程实现自动化。PRISM 将大语言模型、基于物理属性的仿真与实验室自动化硬件结合起来，为自驱动实验室奠定了基础。此类系统有望加速科学发现，同时减少协议执行中的人为错误，从而使实验结果更加可靠且可重复。随着这些技术日趋成熟并得到更广泛的应用，它们可能从根本上改变实验室研究的开展方式，使科学家的角色从手动执行转向更高层次的实验设计与结果解释。

作者贡献

构思：R.S., A.R.;**经费获取：**R.S., I.F., A.R.;**研究：**B.H., P.S., R.B., R.L., C.S., R.W., T.B.;**方法：**B.H., P.S., R.B.;**资源：**R.S., A.R.;**软件：**B.H., P.S., R.B.; **撰写——初稿：**P.S., R.B.;**撰写——审阅与编辑：**P.S., R.B., I.F., A.R.;

利益冲突

作者声明不存在任何利益冲突。

数据可用性

PRISM 的代码和提示可在 https://github.com/ramanathanlab/PRISM 获取。

致谢

我们感谢美国能源部生物与环境研究（BER）和先进科学计算研究（ASCR）项目的支持，资助编号为 DE-FOA-0003600。本文写作过程中使用了 Anthropic 的 Claude、Google 的 Gemini 和 OpenAI 的 GPT 模型的多个版本来协助撰写文本。

参考文献

1 S. Steiner, J. Wolf, S. Glatzel, A. Andreou, J. M. Granda, G. Keenan, T. Hinkley, G. Aragon-Camarasa, P. J. Kitson, D. Angelone 和 L. Cronin，《科学》，2019年，**363**，eaav2211。2 Transcriptic / Strateos，*Autoprotocol*，http://autoprotocol. org，2025年，访问日期：2025年9月21日。3 J. Vrana, O. de Lange, Y. Yang, G. Newman, A. Saleem, A. Miller, C. Cordray, S. Halabiya, M. Parks, E. Lopez, S. Goldberg, B. Keller, D. Strickland 和 E. Klavins，《合成生物学》，2021年，**6**，ysab006。4 Y. Zhang, X. Chen, B. Jin, S. Wang, S. Ji, W. Wang 和 J. Han，*arXiv* 预印本 *arXiv:2406.10833*，2024。5 M. C. Ramos, C. J. Collison 和 A. D. White，*Chemical Science*，2025年，**16**，2514–2572。6 S. Ren, P. Jian, Z. Ren, C. Leng, C. Xie 和 J. Zhang，*arXiv* 预印本 *arXiv:2503.24047*，2025年。7 A. M. Bran, S. Cox, O. Schilter, C. Baldassari, A. D. White 和 P. Schwaller，《自然·机器智能》，2024年，**6**，525–535。8 D. A. Boiko, R. MacKnight, B. Kline 和 G. Gomes，《自然》，2023，**624**，570–578。9 T. Song, M. Luo, X. Zhang, L. Chen, Y. Huang, J. Cao, Q. Zhu, D. Liu, B. Zhang, G. Zou, F. Zhang, W. Shang, J. Jiang 和 Y. Luo，《美国化学会志》，2025年，**147**，12534–12545。10 Y. Ruan, C. Lu, N. Xu, Y. He, Y. Chen, J. Zhang, J. Xuan, J. Pan, Q. Fang, H. Gao, X. Shen, N. Ye, Q. Zhang 和 Y. Mo，《自然·通讯》，2024年，**15**，10160。11 O. O' Donoghue, A. Shtedritski, J. Ginger, R. Abboud, A. Ghareeb 和 S. Rodriques，自然语言处理实证方法会议，2023，页码 2676–2694。12 T. Inagaki, A. Kato, K. Takahashi, H. Ozaki 和 G. N. Kanda，*arXiv* 预印本 *arXiv:2304.10267*，2023。13 U. B. Karli, J.-T. Chen, V. N. Antony 和 C.-M. Huang，ACM/IEEE人机交互国际会议，2024年，第361–370页。14 S. Jiang、D. Evans-Yamamoto、D. Bersenev、S. K. Palaniappan和A. Yachie-Kinoshita，*SLAS Technology*，2024年，**29**，100134。15 W. Wang、S. Swain、J. Lee、Z. Lin、B. Canales、A. Aljovi´c、Y. Liu、Q. Li、A. Marin-Llobet、M. Liu、Z. Gao、R. Liu、J. R. Alvarez-Dominguez和J. Liu，*bioRxiv*，2025年。16 Y. Zhou、J. Yang、Y. Huang、K. Guo、Z. Emory、B. Ghosh、A. Bedar、S. Shekar、Z. Liang、P.-Y. Chen、T. Gao、W. Geyer、N. Moniz、N. V. Chawla和X. Zhang，*arXiv*预印本 *arXiv:2410.14182*，2024年。17 J. A. Douthwaite、B. Lesage、M. Gleirscher、R. Calinescu、J. M. Aitken、R. Alexander和J. Law，*FrontiersinRobotics and AI*，2021年，**8**，758099。18 Y. Su、X. Wang、Y. Ye、Y. Xie、Y. Xu、Y. Jiang和C. Wang，*ChemicalScience*，2024年，**15**，12200–12233。19 N. Yoshikawa、M. Skreta、K. Darvish、S. Arellano-Rubach、Z. Ji、L. Bjørn Kristensen、A. Z. Li、Y. Zhao、H. Xu、A. Kuramshin、A. Aspuru-Guzik、F. Shkurti和A. Garg，自主机器人，2023，**47**，1057–1086。20 K. Darvish、M. Skreta、Y. Zhao、N. Yoshikawa、S. Som、M. Bogdanovic、Y. Cao、H. Hao、H. Xu、A. Aspuru-Guzik、A. Garg和F. Shkurt，*Matter*，2025年，**8**，101897。21 OpenAI，*GPT-5*系统卡，https://cdn.openai.com/gpt-5-system-card.pdf，2025年，大语言模型。22 Anthropic，系统卡附录：*Claude Opus 4.1*，https://www.anthropic.com/claude-opus-4-1-system-card，2025年，大语言模型。23 Anthropic，系统卡：*Claude Sonnet 4.5*，https://www.anthropic.com/claude-sonnet-4-5-system-card，2025年，大语言模型。24 Gemini Team、Google，*Gemini 2.5*：以高级推理、多模态、长上下文和下一代智能体能力开拓前沿，https://storage.googleapis.com/deepmind-media/gemini/gemini_v2_5_report.pdf，2025年，技术报告。25 Google DeepMind，*Gemini 2.5 Flash-Lite*模型卡，https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-2-5-Flash-Lite-Model-Card.pdf，2025年，大语言模型。26 R. Lewis、T. Ginsburg、D. Ozgulbas、C. Stone、A. Stroka、A. Cleary、I. Foster和N. Paulson，*Journal of OpenSourceSoftware*，2025年。27 R. Vescovi、T. Ginsburg、K. Hippe、D. Ozgulbas、C. Stone、A. Stroka、R. Butler、B. Blaiszik、T. Brettin、K. Chard、M. Hereld、A. Ramanathan、R. Stevens、A. Vriza、J. Xu、Q. Zhang和I. Foster，*Digital Discovery*，2023，**2**，1980–1998。28 NVIDIA，*OpenUSD*的*Omniverse*平台，https://www.nvidia.com/en-us/omniverse/，2024年，访问日期：2025-10-14。29 Opentrons Labworks Inc.，*Opentrons OT-2*自动化液体处理机器人，https://opentrons.com/ot-2，2024年，访问日期：2025-10-14。30 Precise Automation Inc.，*PF400*协作型*SCARA*机器人，

https://preciseautomation.com/products/pf400/，2024年，访问日期：2025-10-14。31 Hidex Oy，*Hidex Sense* 微孔板读数仪，https://www.hidex.com/instruments/hidex-sense/，2024年，访问日期：2025-10-14。32 Azenta Life Sciences，自动微孔板封膜机，https://www.azenta.com/products/automated-plate-sealer，2024，访问日期：2025-10-14。33 Azenta Life Sciences，自动微孔板揭膜机，https: //www.azenta.com/products/automated-plate-peeler，2024，访问日期：2025-10-14。34 D. A. Boiko、R. MacKnight 和 G. Gomes，*arXiv* 预印本 *arXiv:2304.05332*，2023。

A 方案规划（阶段1）

本附录介绍第3.1节所述的提示，这些提示供 PRISM 用于生成英语计划，随后将计划转换为适用于机器人的聚合酶链式反应实验方案步骤。其中包括用于开放式生成和受约束生成的提示，并采用多智能体和推理模型方法。

A.1 多智能体框架

A.1.1 聚合酶链式反应的受限提示

WebSurfer 智能体提示词

你是 PRISM 工作流中的第一个智能体（WebSurfer），负责全面收集 Luna Universal qPCR Master Mix实验方案的详细信息。

你的目标是获取 Luna Universal qPCR Master Mix实验方案并提取:
– 试剂清单和体积 – 分步液体处理指令 – 热循环条件（3 步版
本） – 使用 Hidex读数仪进行最终吸光度测量。

你的输出应为清晰的中文指令，结构如下:
1. 材料 2. 液体处理步骤 3.
热循环步骤 4. 吸光度读数

避免进行熔解曲线分析，并明确写出所有数字（不使用范围） .

方案规划器提示

你是 PRISM 工作流中的第二个智能体（方案规划器），接替 WebSurfer智能体开展工作。你的职责是将 WebSurfer 收集的常量转化为一套完整、可执行的 OT-2 液体处理和热循环实验方案，用于 Luna 通用 qPCR 主混合液。

试剂板布局（反应板）A1–A2：Luna 通用
qPCR 主混合液 C1–C4：无核酸酶生物水 D1：正
向引物 E1：反向引物 H1：模板DNA

常量
– 反应孔（A1、A3）包含模板DNA。– 对照孔（A2、A4）不含模板DNA。– 体积（微升）： qPCR 混合液 20、
生物水 5、正向引物5、反向引物 5、模板DNA 5。***注意：对照孔中应使用生物水补足至 40 微升反应体积 -
3 个热循环步骤

输出格式
使用通俗英语按以下确切格式列出所有步骤:
‘X.) 将 ［体积］的 ［试剂］ 从反应板上的 ［板上孔］ 转移到 ［最终板上的目标孔］ ，并混匀 3 个循环。 ［吸头操作 - 弹出吸头］’

完成上述步骤后，追加：‘N.) 热循环： ［温度］为 ［持续时间］’

确保步骤逻辑顺序合理、各孔步骤明确，并可直接用于 OT-2 执行。

评审智能体提示

你是 PRISM 工作流中的第三个代理（Critique），接替方案规划器。你的职责是审查 Planner 为 Luna 通用 qPCR 主混合液生成的 OT-2 实验方案，并提供具体、可执行的反馈，以纠正其中任何物理、逻辑或格式错误。

Yo你的目标是分析机器人代理生成的聚合酶链式反应实验方案，并检查是否存在 ，

– 确认试剂体积和孔位符合以下规格：- 体积（微升）：qPCR 混合液 20、生物水 5、正向引物 5、反向引物 5、模板DNA 5。*** 注意：应使用生物水将对照孔中的反应体积补足至 40 微升 – 反应孔（A1、A3）包含模板DNA；对照孔（A2、A4）不包含模板DNA。– 存在 3 个热循环步骤。– 每个转移步骤均严格遵循以下格式：‘X.) 将 [体积]的 [试剂]从反应板上的微孔板] 的 [孔转移至最终板上的 [目标孔] ，并进行 3 次混匀循环。 [吸头操作 - 弹出吸头]’ – 所有试剂源孔均符合试剂板布局：A1–A2 = Luna Master Mix、C1–C4 = 生物水、D1 = 正向引物、E1 = 反向引物、H1 = 模板DNA。– 步骤顺序合理，且适用于 OT-2。

请以简洁、编号的形式列出任何格式错误、体积错误、源孔错误或不必要的步骤。

验证代理提示词

你是 PRISM 工作流程中的第四个智能体（Validator），位于 Critique智能体之后。你的目标是验证最终的 qPCR 实验方案准确无误、已得到全面修正，并可供 OT-2 执行。

你的目标是确认最终的聚合酶链式反应实验方案准确、格式正确，并可在 OT-2 系统上执行。

验证标准是：
– 所有试剂体积和孔位分配均与常量一致。– 存在 3 个热循环步骤。– 每个步骤均遵循所需格式：‘X.) 将 [体积的] [试剂]从 [微孔板上的] 板上孔转移至反应板上的 [] 目标孔，并进行 3 个混匀循环。 [吸头操作 - 弹出吸头]’ – 试剂源孔必须严格对应：A1–A2 = 为 Luna Master Mix，C1–C4 = 为生物水，D1 = 为正向引物，E1 = 为反向引物，H1 = 为模板DNA。– 顺序合理、移动最少，并纳入所有 Critique 反馈。

仅返回最终更正并验证过的实验方案，其格式应可供 OT-2机器人直接执行。

A.1.2 面向聚合酶链式反应的开放式提示

WebSurfer 智能体提示词

你是 WebSurfer智能体，是 PRISM 科学工作流的一部分。该工作流使用多智能体框架来生成可由 OT-2执行的 Luna® Universal qPCR方案。

目标：从供应商处获取官方的 Luna 通用 qPCR 主混合液实验方案，并仅提取在 OT-2上运行最简定量聚合酶链式反应所需的内容。

请使用通俗英语，以纯文本形式返回一份包含以下部分的单一、自包含报告：

常量（唯一真实来源）
- 每个反应的组分体积（µL）：Luna Master Mix、无核酸酶水、正向引物、反向引物、模板DNA。- 总反应体积（µL）。- 可接受的反应体积范围以及任何关于缩放的说明。- 荧光染料/读出信号及兼容的检测说明（如有注明，包括通道/滤光片）。- 与自动化相关的混匀/操作注意事项（例如建议的混匀方式、避免气泡）。- 热循环程序必须严格表示为三个步骤（初始步骤；包含温度、持续时间和循环次数的循环步骤；最终步骤）。仅使用°C和s；不得使用范围。

材料
- 试剂、实验室器具、仪器。

液体处理指南
- 会影响机器人转移、混匀或添加顺序的供应商备注。

热循环步骤（3 步测试变体）- 直接根据供应商资料整理出 3 步版本（不要自行编造温度或时间）。

荧光读数
- 仪器与染料兼容性及读数通道。如果供应商未说明，请原样写入“供应商未说明。”

规则：
- 明确写出每个数值（微升、°C、s）。不要使用范围。- 如果供应商未提供某项信息，
请写“供应商未说明。”- 除非供应商明确说明，否则不要包含熔解曲线分析。

方案规划器提示

你是方案规划代理，属于 PRISM 科学工作流的一部分。该工作流采用多智能体框架，旨在生成可由 OT-2 执行的 Luna® Universal qPCR 实验方案。你的输入是上方的 WebSurfer 报告。

GOAL
请为 Opentrons OT-2 制定一份完整、物理上可行且生物学上正确的 Luna qPCR 方案，供下游代理使用。

你必须
1) 分配试剂孔和样品孔。2) 设计试剂板（源板）和最终板（PCR反应）的布局。3) 生成完整枚举且格式严格的
OT-2 液体处理步骤。4) 使用 WebSurfer 的常量，准确包含 3 个热循环步骤（初始、循环、最终）。5) 在
Hidex 上进行最终荧光读数。

移液工作站与实验室耗材（请明确声明）
- 最终板（PCR反应）：槽位 1 - 试剂板（源）：注明槽位，务必明确 - 吸头架：注明槽位，务必明确 - 移液
器：单道 P20 和/或 P300；选择适用于该实验方案的最小规格 - 每次转移都使用新的吸头

体积与准确性规则
- 将 WebSurfer 的常量视为事实依据。- 最低准确转移体积为 = 5 微升。如果指定的单次转移体积为 < 5 微升（例如，3 微升正向引物或 3
微升反向引物），则该转移不准确。尽量将最低体积保持在 5 微升。

- 按比例缩放所有组分，同时保持浓度符合要求（例如，保留主混合液的倍数） .
- 如果更改总反应体积（例如，改为 40 微升），请确认并说明浓度仍符合常量的要求。

- 在试剂板上配制每种源，增加 >=10% 的超量余量，并展示计算过程。
- 设计试剂板时，请注意微孔板上每个孔的工作体积（例如，96孔PCR板的每个孔可容纳100ul体积；如果需要 >= 100ul的某种试剂，则应在微孔
板上准备2个装有该试剂的孔）。

最终反应板设计
- 恰好设置2个测试孔（包括模板DNA）和2个对照孔（不包括模板DNA）。请明确标识（例如，A1、A3=测试；A2、A4 = 对照）。

- 在最终反应板上将各孔分隔开，以避免污染。

严格步骤格式（不得使用简写，不得使用“对孔重复操作”之类的表述）
X.) 从反应板上的 [板上孔] 转移 [体积]的 [试剂]至最终板上的 [目标孔] ，混匀3个循环。 [吸头操作 - 弹出吸头]

输出结构（仅输出以下内容） 1) 材料与板图 2) 液体处理步骤（严格按格式） 3) 热循环步骤（3 个步骤） 4) 荧光读取（Hidex；如果
WebSurfer 指定了通道，则注明通道；否则写“供应商未指定通道。”）

批评与验证器提示词

你是合并后的 Critique + Va 验证代理，是 PRISM 科学工作流的一部分，该工作流使用多智能体框架 t o
生成可由 OT-2 执行的 Luna U niversal qPCR 实验方案。

输入：
- Protocol Planner 的输出（材料与板图、液体处理步骤、热循环、荧光读取）- WebSurfer 报告（常量——将其视为事实来源）

CRITIQUE — 检查项（列出问题并给出具体修正方案）

A）与常量的一致性
1. 每反应组分体积与 WebSurfer 常量一致，或按比例缩放且保持浓度不变（例如，维持 2× 主混合液倍数）。

2. 每孔体积总和等于声明的总量。
3. 对任何缩放都明确说明理由，并确保引物/模板浓度有效。

B) 板位图、超量余量与容量
4. 试剂板列出所有源，并包含 >=10% 的超量余量，同时展示计算过程。
5. 遵守试剂孔容量限制：如果 96孔PCR板中任何试剂的总量需要 >=100 微升，则必须分配到 >=2 个源孔中，并明确标注对应关系。

6. 最终板恰好包含 2 个测试孔（包括模板DNA）和 2 个对照孔（不包括模板DNA），且所选孔彼此不相邻，以降低污染风险。请明确标出这些孔（例如：A1、A3 = 为测试孔；A2、A4 = 为对照孔）。

C) 最低转移量、枚举与吸头卫生
7. 单次转移量不得为 < 5微升。如果 WebSurfer 常量指定了低于 5 微升的组分（例如 3 微升引物），请确认按比例缩放或采用其他合规方案，以确保浓度有效。8
. 每条液体处理指令都完整列出——不使用“对所有孔重复”这类简写 .
9. 接触引物或模板时使用新吸头；移液器的选择应符合实验方案要求（例如 P20 或 P300）。

D) 格式、实验室器具与实验台
10. **严格遵守行格式规范（必须完全一致）： ** X.) 将 [volume]of [reagent] 从 [板上孔]（反应板上）转移至 [目标孔] （最终板上），并进行 3 次混匀循环。 [吸头操作 - 弹出吸头]- 体积单位必须为微升；试剂名称必须与材料/板位图一致。- 源孔必须存在于试剂板上；目标孔必须存在于最终板上。

11. 台面声明明确且一致：最终板 = 槽位 1；试剂板 = 槽位 2；吸头架 = 槽位 7–8（或明确注明其他方案）。

E) 热循环与 Hidex
12. 恰好包含 3 个热循环步骤，温度（°C）和时间（s）须与常量一致（不得使用范围）。
13. 必须包含 Hidex 读取步骤；若常量中指定了通道，则须注明该通道，否则写明“供应商未指定通道。”

如果所有检查均通过：返回“未发现问题。”否则：列出问题，并提供修正后的行和/或修正后的部分。

VALIDATOR — 最终输出

应用修正后，仅返回最终修正后的实验方案，并包含以下部分：
1) 材料与板图
2) 液体处理步骤
3) 热循环步骤
4) 荧光读取

严格的液体处理行格式（必须完全一致）
X.) 将 [volume]的 [reagent]从反应板上的 [板上孔] 转移到最终板上的 [目标孔] ，混匀 3 个循环。 [吸头操作 - 弹出吸头]

- 将占位符替换为具体值（例如“40 微升”、“Luna 通用 qPCR 主混合液”、“A1”等）。
- 请勿使用简写或“重复步骤”之类的表述。

A.1.3 细胞绘画提示

WebSurfer 智能体提示词

你是一位能力出众的代理，负责浏览互联网，全面收集有关 PhenoVueTM 细胞绘画试剂盒及其实验方案的详细信息。你的目标是提取并清晰呈现：

PhenoVueTMCell Painting Kit 中所含材料的详细清单。使用试剂盒试剂配制染色溶液的分步说明。细胞制备、染色、固定、透化和洗涤的实验步骤。关于试剂体积、孵育时间和操作条件（例如温度、光敏性）的具体指南。成像参数建议（例如荧光通道的显微镜设置）。确保所有信息均以清晰易懂的中文呈现。避免使用专业术语，并突出对机器人自动化至关重要的细节。

方案规划器提示

你是一名高级智能体，负责将科学上准确的细胞绘画实验方案转换为完全兼容 OT-2 的液体处理协议。

你的职责：

1。使用以下内容将每个步骤转换为适用于 OT-2 的格式：- 具体的 OT-2 槽位编号（1–11）- 与 OT-2 兼容的有效实验器材类型（例如，12 列储液槽、试管架、96 孔板）- 实验器材中的孔位（例如，A1、B2、C3）- 混匀循环（始终为 3 次）- 吸头操作（“[Tip action: eject]”）

2. 实验室器具与标记：
- 为每件实验室器具（例如 96孔板或储液槽）指定一个**清晰、通用的标签**，例如“反应板”、“染色储液槽”或“DNA 试管架”。

- 这些标签必须各不相同，并在整个实验步骤中**始终保持一致地引用**。
- 如果使用多个相同实验器材类型的器具（例如，两个 96 孔板），请为它们指定不同的标签和槽位数字。

3. 试剂和材料追踪：
- 在实验步骤之前，生成**试剂清单**和**材料清单**，并标明分配给它们的 OT-2 槽位和孔位。

- 在材料部分，列出每件实验室器具及其：- 槽位编号 - 实验器材类型（例如：12通道储液槽、96孔微孔板）- 分配的标签（例如：“反应板”）

4. 试剂体积管理：
- 如果某种试剂在多个步骤中使用，则重复使用同一个孔，直到其中的体积耗尽。- 需要时，切换到该试剂下一个可用的孔或列。- 跟踪累计用量，确保剩余体积充足。

所有移液操作的格式（必填）：

X.) 将 [volume]的 [reagent] 从 [labware label]孔 [source well] 转移至 [labware label] 孔 [destination well]，并进行 3 次混匀循环。 [吸头操作：弹出吸头]

示例：

1.) 将 100 微升固定缓冲液从染色储液槽 A3 孔转移至反应板 B1 孔，并进行 3 次混匀循环。 [吸头操作：弹出吸头]

2.) 将 50 微升 1X 汉克斯平衡盐溶液从洗涤储液槽 C2 孔转移至反应板 D2 孔，并进行 3 次混匀循环。 [吸头操作：弹出吸头]

注意事项：
- 请在步骤中注明实验器材类型和标签（例如“试管架”、“储液槽”、“反应板”）。- 步骤中不要直接提及槽位，而应使用材料清单中指定的标签。- 所有步骤都必须编号、可供机器人执行且完整自足。

你的最终输出必须包括：1. 试剂清单（试剂名称 + 孔位 +实验器材标签）

2. 材料清单（槽位 + 实验器材类型 + 标签）
3. OT-2液体处理步骤的编号列表（可供机器人执行且格式规范）

此实验方案必须在生物学上准确、物理上可行，并且能够在 OT-2 上完整执行。

评审智能体提示

你的任务是批判性地分析机器人代理编写的细胞绘画实验方案。你的目标是确保该实验方案：

- 符合格式要求 - 逻辑正确 - 提高机器人执行的效率并进行优化 - 正确使
用实验器材标签并追踪试剂/材料

步骤格式：

每条液体处理说明都必须遵循以下确切格式：

X.) 将 [体积]的 [试剂] 从 [实验器材标签]孔 [源孔] 转移至 [实验器材标签] 孔 [目标孔]，并进行 3 次混匀循环。 [吸头操作：弹出吸头]

示例：
1.) 将 50 µl 的 1X HBSS 从染色储液槽 A1 孔转移至反应板 B3 孔，并进行 3 次混匀循环。 [吸头操作：弹出吸头]

无效示例：
- “加入 100 µl 的 PBS”（缺少结构和标识符） - “将细胞转移至新的微孔板”（不具体，
缺少体积或位置） - “从微孔板 A1 转移至微孔板 B1”（没有试剂名称和实验器材标签）

你的评审任务：

1. **试剂和材料清单** - 实验方案是否包含完整的**试剂清单**和**材料清单**？ - 试剂是否已正确分配到
带标签的实验室器具中的孔位？ - 材料是否已分配到 OT-2 槽位，并使用有效的实验室器具名称和唯一标签？

2. **步骤验证**
- 所有步骤是否均已正确编号并格式化？ - 是否包含试剂名称、孔位和实验器材标签，且前后一致？

3. **实验方案逻辑与流程** - 步骤顺序是否符合细胞绘画的生物学逻辑（例如，先染色再固定）？ - 孵育、封板和揭封
指令的安排是否合理？

4. **优化建议**
- 是否存在不必要的重复操作或冗余转移？ - 实验方案能否提高移液效率或减少吸
头使用？ - 切换到另一个孔之前，是否合理地重复使用试剂？

重要提示：
- 格式必须符合机器人标准，且在生物学上合理。 - 使用项目符号或编号列表清楚
地指出问题。 - 提出改进建议时，请使用机器人操作风格的示例。

验证代理提示词

你是 由机器人代理生成的 Cell Painting协议的最终验证关卡。

您必须确认实验方案：- 兼容机器人 - 逻辑
和结构合理 - 使用有效的实验室器具和格式

所需结构：

1. 一份包含以下内容的**试剂清单**：-
试剂名称 - 指定的实验器材标签 - 孔位
（例如，A1、B2）

2. **材料清单**，包括： - OT-2 槽位编号 - 实验器材类型（例如，12列
储液槽、96孔板） - 指定标签（例如，染色储液槽、反应板）

3. **编号的移液步骤**列表，每个步骤均遵循以下格式：

X.) 将 [体积]的 [试剂] 从 [实验器材标签]孔 [源孔] 转移到 [实验器材标签] 孔 [目标孔]，混匀 3 个循环。 [吸头操作：弹出吸头]

示例：
1.) 将 100 微升固定缓冲液从染色储液槽 A3 孔转移到反应板 B1 孔，混匀 3 个循环。 [吸头操作：弹出吸头]

无效：
- 步骤缺少实验器材标签 - 指令含糊（例如：“向微孔板中加入缓冲液”）
- 步骤缺少试剂名称或孔位 - 步骤引用槽位编号而非实验器材标签

您的验证任务：

1. **格式规范** - 每个步骤都必须遵循上述正确格式。 - 步骤中不应出现槽位编号——
只能使用实验器材标签。

2. **实验方案逻辑**
- 步骤顺序应符合 CellPainting实验方案（试剂准备 → 染色 → 固定 → 洗涤 → 成像） - 确认包含封板、揭封和孵育指令

3. **一致性检查**
- 步骤中引用的每种试剂和材料都必须出现在试剂/材料列表中。- 试剂必须始终使用同一个孔，直至耗尽，然后转移到下一个
可用孔。

输出指令：
- 如果所有步骤和结构均有效，则返回完整实验方案。- 如果任何要求未满足，则
返回：

- 实验方案已拒绝：步骤格式、逻辑或结构无效。请重新生成 e.

不要返回部分结果或解释。仅进行验证或拒绝。

A.2 单智能体

A.2.1 约束提示

你是一个能力出众的推理模型，负责为 Luna 通用 qPCR 主混合液生成并验证一份完整、可执行的 qPCR 实验方案，其中包括反应孔和对照孔。

你的目标是：
1. 收集准确的试剂信息和热循环详情。 2. 为 Opentrons OT-2 机器人生成逐步的液体处理指令。 3. 严格
检查自己的输出，确保准确性、格式规范和逻辑可行性。 4. 返回经过最终修正和验证、可供执行的实验
方案。

试剂板布局（反应板） A1–A2：Luna®
UniversalqPCRMaster Mix C1–C4：无核酸酶
生物水 D1：正向引物 E1：反向引物 H1：模板
DNA

所需规格
- 反应孔：A1 和 A3（含模板DNA）
- 对照孔：A2 和 A4（不含模板DNA）
- 体积（微升）：
  - qPCR预混液 – 20
  - 生物水 – 测试用 5，对照用 10
  - 正向引物 – 5
  - 反向引物 – 5
  - 模板DNA – 5（仅用于 A1、A3）
- 总计：**18 个液体处理步骤** + **3 个热循环步骤**
- 所有转移均使用 20 µL移液器。

步骤格式
每个液体处理步骤都必须严格遵循：
‘N.) 将 [volume]的 [reagent] 从反应板上的 [板上孔] 转移至最终板上的 [目标孔] ，并进行 3 次混匀循环。 [吸头操作 - 弹出吸头]‘

热循环步骤必须遵循：‘N.) 热循环： [温度]持续 [持续时间]。
‘

最终输出前自检
生成实验方案后，请检查：
- 所有试剂体积和孔位映射均正确。- 反应孔中包含模板DNA；对照孔中不包含模板DNA .
- 恰好包含18个液体处理步骤和3个热循环步骤。- 每个步骤均严格遵循所要求的格式。- 步骤顺序合理，且适用于OT-2执行。

输出格式
仅提供最终验证通过的实验方案，并按以下结构编排：

1. 材料 2. 液体处理步骤 3. 热循环步骤 4. 吸光度读数（使用Hidex酶标仪）

避免进行熔解曲线分析。明确列出所有数字（不要使用范围）。最终答案中不得包含评论或推理过程。

A.2.2 开放式提示

你是一个能力出众的推理模型，负责使用 Luna 通用 qPCR 主混合液生成一份完整且可由 OT-2 执行的 qPCR 实验方案。

你的任务是：(A) 查找权威的实验方案参数；(B) 将其转换为适用于 OT-2、且符合以下实验室配置的实验方案。仅输出最终验证通过的实验方案。生成的实验方案必须能够在 OT-2 上实际执行，并符合生物学要求（维持正确的浓度，避免抑制性条件，并尽量减少交叉污染）。

目标
1) 获取官方 Luna通用 qPCR预混液实验方案，并提取所需常量：组分列表、每个反应的体积、热循环温度/时间以及染料兼容性。2) 生成符合“实验设置与约束条件”的明确、编号 OT-2 液体处理和热循环步骤。3) 加入两个测试反应和两个对照反应，并将其放置在相距较远的位置，以最大限度减少交叉污染。4) 最后使用 Hidex酶标仪进行荧光读取。

实验设置与约束条件
- OT-2 实验台：**最终PCR板**位于**槽位 1**，**试剂板**位于**槽位 2**。- 两者均使用**96孔PCR板**。
- **每孔工作体积：**100 微升。如果试剂总用量超过该体积，则将其分配到多个源孔中，并在板位图中注明。

- **单次吸液/分液的最小体积：**5 微升——不得进行低于此体积的转移。
- **正向引物和反向引物必须分开处理。** 每种引物都需要针对每个反应孔单独设置一个转移步骤（>= 5 µL）；

- **测试孔的反应体积：**40 µL（根据 Luna 常量得出）。- **对照孔：**设置相同，但不加入模板 DNA（改用合适的稀释液）。- **热循环：**总共恰好 3 个步骤，使用 Luna 官方规定的阶段温度/持续时间。- **荧光读取：**Hidex酶标仪，SYBR/FAM 通道。- 不要使用“重复步骤”；明确列出每一次转移。- 请自行分配源孔，并在板位图中报告。- 在试剂表中加入任何必要的超量余量/死体积调整。

格式规则
仅按顺序列出以下六个部分，不要添加任何说明：
1. 材料 2. 板位图 3. 试剂配制（如有使用，请包含超量余量计算）
4. 液体处理步骤 5. 热循环步骤 6. 荧光读取

每条液体处理指令都必须遵循以下确切格式：
N.) 转移 [体积] of [reagent]来自试剂板上的 [源孔]，转移至最终板上的 [目标孔] ，混匀 3 次次。 [吸头操作 – 弹出吸头]

Thermocycling format: N.) 热循环： [温度 °C] 持续 [时间 s]。

荧光读取（单行）： N.) 荧光读取：Hidex酶标仪，SYBR/FAM通道。

规划要求（内部使用——勿打印）
- 根据官方 Luna 常量计算各组分体积，使测试总体积为 = 40 微升。
- 选择两个 测试孔和两个空间间距最大的对照孔；在微孔板示意图中列出它们
p. - 如果任何试剂的总用量超过 100 微升，请分配到多个源孔中，并记录孔位映射。
- 确认每次吸液至少为 5 微升、试剂与孔位的映射正确、模板的纳入或排除正确，并符合格式要求。

- 所有步骤都必须能够使用标准 OT-2移液器执行（每次转移使用新的吸头，并正确混匀）。- 输出必须恰好包含上述六个部分，不得包含其他内容。

A.2.3 细胞绘画提示

你是一个高层次推理模型，负责使用 PhenoVueTM 细胞绘画试剂盒生成一套完整、可执行的细胞绘画实验方案，该方案专为 Opentrons OT-2 自动化而设计。

你将以整体方式思考和行动——而非作为多个智能体——规划、推理并输出最终经过验证、可供机器人执行的实验方案。

目标：

生成一份可直接用于 OT-2 的完整实验方案，确保其在生物学上正确、在物理上可行，并采用适合机器人执行的格式。

任务

1. 理解任务
- 回忆或推断 PhenoVueTM 细胞绘画试剂盒的所有必要细节：试剂、染色和固定溶液、孵育步骤以及成像读出。

- 如果没有确切数据，请运用推理，为试剂体积、孵育时间或操作
步骤设定合理且明确说明的假设。

2. 设计机器人工作流程
- 将 OT-2 实验室器具分配到槽位（1–11），并清晰标注（例如“反应板”“染色储液槽”“洗涤储液槽”）。

- 按照逻辑上的细胞绘画工作流规划操作顺序：细胞接种 → 染色 → 固定 → 透化 →洗涤 → 成像。- 确保所有转移、孵育以及封板/揭封操作都遵循切实可行的机器人操作顺序。

3. 生成结构化输出
最终实验方案必须包含：- 试剂清单：试剂名称 + 实验器材标签 + 孔位置 - 材料清单：OT-2 槽位 +实验器材类型 +分配的标签 - 液体处理步骤：编号并严格按照以下格式编写：

X.) 将 [体积的 [试剂] 从反应板上的 [板上孔]转移到最终板上的 [目标孔]，并进行 3 次混匀循环。 [吸头操作 - 弹出吸头]

4. 推理并验证
- 检查你自己的输出是否符合以下要求：
  - 结构正确（试剂、材料、步骤）
  - 列表与步骤之间的一致性
  - 符合逻辑的生物学顺序与可行性
  - 格式正确（步骤中不使用槽位编号，仅使用标签）——如果发现不一致，请在最终输出前修正。

5. 仅交付最终且经过验证的实验方案——不要描述你的推理过程。——仅输出最终、经过全面验证且可供 OT-2 执行的实验方案。

预期输出格式
1. 试剂清单 2. 材料清单 3. 液体处理步骤（编号） 4. 孵育/封板/成像说明（如适用）

B 协议生成（第 2 阶段）

B.1 初始协议生成提示

以下是第 3.2 节中使用的完整提示词，用于让 PRISM 将英文方案转换为适用于聚合酶链式反应实验方案的机器人兼容步骤。内容涵盖实验室中可用的机器人、所需的输出格式，并以待转换的具体工作流程结尾。

YAML 工作流文件文档

模块和操作

模块：‘biopf400‘
用途：机器人转移系统（主板处理机器人）

访问：
此模块　　没有自己的位置，而是负责在所有其他位置之间进行转移　　s.

限制：
- ‘转移’要求源位置上有微孔板 - ‘转移’要求目标位置可用/为空 - ‘转移’只能在板旋转方向相同的位置之间进行（均为窄或均为宽） - 如果转移需要改变板方向（从窄到宽或从宽到窄），则必须先经过交换站作为中间步骤

动作：
- **‘transfer‘**：在不同位置/站点之间转移微孔板，并可配置定位方式 - **参数**：

  - ‘source‘：源位置标识符 - ‘source_approach‘：源端取板的接近路径标识符 - ‘source_plate_rotation‘：源位置处的板方向（‘宽‘或‘窄‘） - ‘target‘：目标位置标识符 - ‘target_approach‘：目标端放置的接近路径标识符 - ‘target_plate_rotation‘：目标位置处的板方向（‘宽‘或‘窄‘）

模块：‘ot2bioalpha‘
用途：OT-2 液体处理机器人

访问：
位置：
- ‘ot2bioalpha_deck1_wide‘ - OT-2机器人甲板位置1（宽）

接近路径： - ‘safe_path_ot2bioalpha‘ - OT-2机器人的安全接近路径

限制：
- ‘run_protocol‘ 只能在微孔板和耗材正确装载到实验台上后执行

操作：
- **‘run_protocol‘**：使用指定的实验方案文件执行液体处理协议　　le
  - **参数**：- ‘文件‘：包含实验方案文件的字典 - ‘实验方案‘：OT-2 实验方案文件的路径（例如，‘payload.ot2_protocol‘）

模块：‘bio_sealer‘
用途：封板站

访问：
位置：
- ‘封板机_托架‘ - 封板站托架（窄）

接近路径：
- ‘safe_path_sealer‘ - 封板工位的安全接近路径

限制条件：
- 只有封板机载台上有微孔板时，才能执行“封板”

操作:
- **‘封板‘**：使用粘合膜或热封膜（通常）封闭微孔板 - **参数**：无（空参数）

模块：‘bio_biometra3_96‘ **用途**：用于聚合
酶链式反应的 Biometra 热循环仪

访问:
位置：
- ‘bio_b iometra3_托架‘ - Biometra 热循环仪托架（ 宽)

接近路径：
- ‘safe_path_biometra3‘ - Biometra 热循环仪的安全接近路径

限制:
- 将微孔板放入托架或从托架中取出之前，必须处于‘打开‘状态 - 执行‘close‘d之前，必须先关闭‘run_program‘ - 仅当设备处于关闭状态且托架中有微孔板时，才能执行‘run_program‘ - ‘run_program‘完成后，必须再次‘open‘以取出微孔板

动作:
- **‘打开‘**：打开热循环仪盖 - **参数**：无（空参数） - **‘关闭‘**：关闭热循环仪盖 - **参数**：无（空参数） - **‘run_program‘**：执行预设的热循环实验方案 - **参数**： - ‘program_number‘：指定要运行的已存储程序的整数（例如，‘5‘）

模块：‘bio_peeler‘ **用途**：微孔板揭膜/拆封站

访问：**位置：** - ‘peeler_nest‘ - 揭膜站托位（窄）

接近路径： - ‘safe_path_peeler‘ - 揭膜站的安全接近路径

限制:
- 只有当密封微孔板位于撕膜机托架中时，才能执行‘剥离’

动作：- **‘剥离‘**：从微孔板上移除封板
- ** 参数 **：无（空参数）

模块：‘hidex_geraldine‘ **用途**：用于分析测定/测量的
Hidex酶标仪

访问:
位置：
- ‘hidex_geraldine_high_nest‘ - Hidex酶标仪的高位托架位置（窄）

接近路径： - ‘safe_path_hidex‘ - Hidex酶标仪的安全接近路径

限制:
- 必须处于“打开”状态，才能将微孔板放入托架或从托架中取出 - 执行“检测”_前必须处于“关闭”状态 - 只有在关闭且托架中有微孔板时，才能执行 “ 检测 ”_ - “ 检测 ”_ 完成后，必须再次 “ 打开 ” 才能取出微孔板

操作:
- **‘打开‘**：打开读板仪盖 - **参数**：无（空参数） - **‘关闭‘**：关闭读板仪盖 - **参数**：无（空参数）

- **‘运行_检测‘**：运行指定的检测方案 - **参数**：- ‘检测_名称‘：检测方案的字符串名称（例如，‘"聚合酶链式反应_最终_结果"‘）

交换站
用途：交换站作为模块之间的中间转移点，可在多步骤工作流程期间实现板方向变更和临时存储。

访问权限：

位置：
- ‘exchange_deck_high_wide‘ - 交换站高位（宽） - ‘exchange_deck_high_narrow‘ - 交换站高位（窄） w）

接近路径：
- ‘safe_path_exchange‘ - 交换站的安全接近路径

限制：
- 交换台必须是转移中的源位置或目标位置，绝不能两者兼具
- 所用交换台的方向必须与另一个位置一致，即两者都为宽向或都为窄向
- 通过交换台进行转移时应使用两次转移：第一次以交换台为目标，第二次以交换台为源

YAML 工作流文件格式规范

本节介绍自动化实验室规 程所使用的 YAML 工作流文件的格式和结构要求。

顶层结构

每个 YAML 工作流文件必须包含以下顶层字段，后跟一个“flowdef”部分 n：

‘“yaml
name: [规程名称]
author: [作者/组织]
info: [规程描述]
version: ’[版本号]’

flowdef: # 操作列表请写在这里 “‘

元数据字段

‘名称‘（必填）
- **类型**：字符串
- **用途**：便于人类阅读的实验方案名称
- **示例**：‘"测试实验方案"‘、‘"PCR 扩增工作流程"‘

‘作者‘（必填）
- **类型**：字符串 - **用途**：创建该实验方案的个人或组织名称 - **示例**：‘" Autoprotocol"‘、‘"实验室自动化团队"‘

‘信息‘（必填）
- **类型**：字符串
- **用途**：简要描述此实验方案的用途
- **示例**：‘"由 Autoprotocol 编写的 PCR 验方案"‘，‘"用于基因表达分析的自动化 qPCR 工作流程" ‘实

‘版本‘（必填）
- **类型**：字符串（带引号） - **用途**：用于实验方案跟踪和更新的版本标识符 - **示例**：‘’0.1’‘、‘’1.2.3’‘、‘’2024.01’‘

flowdef 定义（‘flowdef‘）

‘flowdef‘ 部分包含一个有序的操作列表，用于定义工作流执行顺序。

操作结构

‘flowdef‘ 列表中的每个操作都必须采用以下结构：

“‘yaml - 操作： [操作_名称]名称： [人类_可读_描述]模
块： [模块_标识符]参数： [参数_对象] #可选 文件： [
文件_对象] #可选 “‘

操作字段

‘操作‘（必填）
- **类型**：字符串
- **用途**：指定要执行的操作
- **有效值**：可用操作请参阅模块文档
- **示例* *：‘run_protocol‘、‘transfer‘、‘seal‘、‘open‘、‘close‘、‘run_program‘、‘peel‘、‘run_assay ‘

‘name‘（必填）
- **类型**：字符串
- **用途**：以人类可读的方式描述此特定操作的功能
- **示例例**：‘"运行液体处理协议"‘，‘"将目标板从 OT-2 转移至交换站"‘

‘模块‘（必填）
- **类型**：字符串
- **用途**：指定将执行该操作的设备模块
- **有效值**：‘ot2bioalpha‘、‘biopf400‘、‘bio_sealer‘、‘bio_biometra3_96‘、‘bio_peeler‘、‘hidex_geraldine‘

‘参数‘（可选）
- **类型**：对象/字典 - **用途**：包含执行该操作所需的专用参数 - **用法**：需要参数的操作必须提供（例如，转移位置、程序编号） - **示例**：“‘yaml args: source: ot2bioalpha_实验台1_宽 target: exchange_实验台_高_宽 source_接近方式: 安全_路径_ot2bioalpha target_接近方式: 安全_路径_exchange source_微孔板_旋转: 宽 target_微孔板_旋转: 宽 “‘

‘文件‘（可选）
- **类型**：对象/字典
- **用途**：指定操作所需的文件引用
- **用法**：主要用于‘run_实验方案‘操作 s
- **示例**：“‘yaml files: protocol: payload.ot2_protocol “‘

完整示例

“‘yaml 名称: 示例协议 作者: PRISM 信息: 带封板的自动化工作流程 版本: ’1.0’

flowdef:
- 操作：运行_规程名称：执行液体处理规程模块：ot2bioalpha 文件：

实验方案：payload.ot2_实验方案

- 操作：转移 名称：将微孔板从OT-2移动到封板模块 模块：biopf400 参数：源：ot2bioalpha_实验台1_宽型源_接近方式：安全_路径_ot2bioalpha 源_微孔板_旋转：宽 目标：封板机_托架 目标_接近方式：安全_路径_封板机 目标_微孔板_旋转：窄

- 操作：封板 名称：封住反应微孔板 模块：bio_封板机 参数：{} “‘

重要说明

- **YAML语法**：确保缩进正确（建议使用 2 个空格）且 YAML 格式有效 - **操作顺序**：操作按照列出的顺序依次执行 - **空参数**：对于需要 args 字段但不需要参数的操作，使用 ‘args: {}‘ - **注释**：使用 ‘#‘ 添加注释和章节分隔符，以提高可读性

- **字符串引号**：为版本号以及包含特殊字符的字符串加上引号

OT-2液体处理文件格式规范

本文档介绍了用于 Opentrons OT-2 液体处理机器人的 Python 协议文件所需的格式和结构。

顶层结构

每个 OT-2 实验方案文件都必须包含以下组件：

“‘python requirements = {"机器人类型": "OT-2"} from opentrons import 实验方案_api

metadata = { "protocolName": "PCR", # 人类可读的实验方案名称 "author": "PRISM", # 实验方案作者/创建者 "description": "PCR", # 实验方案的简要描述 "apiLevel": "2.12", # Opentrons API 版本 "info": "由 PRISM 编写的 PCR 实验方案", # 其他信息 "name": "PCR", # 实验方案简称 "version": "1.0" # 版本标识符 }

defrun(protocol: protocol_api.ProtocolContext): # 实验方案实现 “‘

协议函数

模块加载 “‘python module = protocol.load_模块("温控模块", "3") 实验台["3"] = module.load_实验室器具("nest_96_wellplate_100ul_pcr_full_skirt")

deck["3"].set_offset(x=1.0, y=1.4, z=5.7)
‘‘‘

标准实验室器具加载 “‘python deck["1"] = protocol.load_labware("nest_96_wellplate_100ul_pcr_full_skirt","1") deck["1"].set_offset(x=0.9, y=-0.5, z=0.6)

‘‘‘

吸头架装载 “‘python deck["7"] =protocol.load_labware("opentrons_96_tiprack_20ul", "7") deck["7"].set_offset(x=0.2, y=1.6,z=-0.8) “‘

移液器装载 “‘python pipettes["left"] = protocol.load_instrument("p20_single_gen2", "left", tip_racks=[deck["7"]]) “‘

标准转移流程

单次转移操作并进行混匀的基本流程：

“‘python pipettes["left"].pick_up_tip() pipettes["left"].well_bottom_clearance.aspirate = 1pipettes["left"].aspirate(20.0, deck["3"]["A1"]) pipettes["left"].well_bottom_clearance.dispense = 1pipettes["left"].dispense(20.0, deck["1"]["B2"]) pipettes["left"].mix(3, 20, deck["1"]["B2"]) pipettes["left"].blow_out() pipettes["left"].drop_tip() “‘

操作分解

1. **‘pick_up_tip()‘**：从吸头架中取出一个新吸头 2. **‘well_bottom_clearance.aspirate‘**：设置吸液时高于孔底的高度（单位：毫米） 3. **‘aspirate(volume, location)‘**：从指定孔中吸取液体 4. **‘well_bottom_clearance.dispense‘**：设置分液时高于孔底的高度（单位：毫米） 5. **‘dispense(volume, location)‘**：将液体分液至指定孔中 6. **‘mix(repetitions, volume, location)‘**：通过反复吸液和分液进行混匀 7. **‘blow_out()‘**：排出吸头中剩余的液体 8. **‘drop_tip()‘**：丢弃用过的吸头

仅混匀操作

无需转移，仅进行混匀：

“‘python 移液器["left"].pick_up_tip() 移液器["left"].孔_bottom_clearance.吸液 = 1移液器["left"].孔_bottom_clearance.分液 = 1移液器["left"].混匀(10, 20, 实验台["1"]["B2"]) 移液器["left"].blow_out() 移液器["left"].drop_tip() “‘

最小可运行示例

“‘python
要求 = {"机器人类型": "OT-2"}
从 Opentrons 导入 protocol_api

元数据 = { "protocolName": "简单转移", "author": "PRISM", "description": "基本液体转移示例", "apiLevel": "2.12",

"info": "演示 minimalOT-2 实验方案结构", "name": "简单转移",
"version": "1.0" }

定义 run(protocol: protocol_api.ProtocolContext): 实
验台 = {} 移液器 = {}

################ # 加载实验室器具 # ################ 实验台["1"] =protocol.load_labware("nest_96_wellplate_100ul_pcr_full_skirt", "1") 实验台["2"] =protocol.load_labware("nest_96_wellplate_100ul_pcr_full_skirt", "2") 实验台["7"] =protocol.load_labware("opentrons_96_tiprack_20ul", "7") 移液器["left"] = protocol.load_instrument("p20_single_gen2", "left", 吸头_racks=[实验台["7"]]) ################### # 执行命令 # ################### # 从源向目标转移 10 微升并混匀 移液器["left"].pick_up_tip() 移液器["left"].well_bottom_clearance.吸液 = 1移液器["left"].aspirate(10.0, 实验台["2"] ["A1"]) 移液器["left"].well_bottom_clearance.分液 = 1移液器["left"].dispense(10.0, 实验台["1"] ["A1"]) 移液器["left"].mix(3, 10,实验台["1"] ["A1"]) 移液器 ["left"].blow_out() 移液器 ["left"].drop_ 吸头 () ‘‘‘

重要说明

- **吸头使用**：每次转移操作都使用新的吸头（开始时 pick_up_tip，结束时 drop_tip）- **孔位引用**：孔位使用字符串表示（例如 ‘"A1"‘、‘"B2"‘）- **注释**：使用 ‘#‘ 添加注释和章节分隔符，以提高可读性 - **顺序执行**：命令按书写顺序执行

PCR工作流程顺序概览

以下是自动化PCR处理实验的高级工作流程步骤。

‘ot2bioalpha‘ OT-2 已准备就绪：96孔板 ‘react_plate‘ 放置在实验台位置 3 的温控模块上，‘dest_plate‘ 放置在实验台位置 1

1.) 将 20 uL 的 qPCR预混液从 react_plate 转移到 dest_plate：A1→B2、A2→B11、A1→G2、A2→G11，混匀 3 个循环。 [吸头操作 – eject]

2.) 将 5 uL 无核酸酶生物水从反应_微孔板转移至目的_微孔板：C1→B2、C2→B11、C3→G2、C4→G11，进行 3 次混匀循环 次。
[吸头操作 – 弹出]
3.) 将 5 uL 正向引物从反应_微孔板 D1 转移至目的_微孔板的 B2、B11、G2 和 G11 孔，进行 3 次混匀循环。 [吸头操作 – 弹出]

4.) 将 5 uL 反向引物从反应_微孔板 E1 转移至目的_微孔板的 B2、B11、G2 和 G11 孔，进行 3 次混匀循环。 [吸头操作 – 弹出]

5.) 将 5 uL 模板DNA从反应_微孔板 H1 转移至目的_微孔板的 B2 和 G2 孔，进行 3 次混匀循环。 [吸头操作 – 弹出]6.) 将 5 uL 无核酸酶生物水从反应_微孔板转移至目的_微孔板：C2→B11、C4→G11，进行 3 次混匀循环。 [吸头操作 – 弹出]

7.) 用移液器设置为 20 uL，在 B2、G2、B11 和 G11 孔中混匀 10X
8.) 封板，为热循环做准备 9.) 执行热循环操作（程序 #5）

10.) 剥离微孔板，为读数做准备
11.) 使用 Hidex酶标仪进行吸光度读数（实验方案 "PCR_Final_Results"）
12.) 分析完成后打开 Hidex板式读数仪盖

你是 PRISM，一款用于科学工作流程的自动化实验方案设计器。
使用提供的参考材 料，创建 PCR 工作流程序列所需的所有 OT-2 Python 文件和工作流程 YAML 文件 e.

将所有相邻的液体转移操作合并为一次在 OT-2 机器人上执行的液体处理协议。为每个 OT-2协议Python文件取一个独特且具有描述性的名称。所有液体处理之外的步骤，包括注释中的内容或标明为手动操作的步骤，都应写入 YAML文件。

B.2 初始协议生成输出

以下是 Claude Sonnet 4.5 对上一节提示词生成的原始输出。请注意，原始输出描述了两个文件（一个 Python 文件和一个 YAML 文件），这些文件已手动提取并用于下游任务。

查看 PCR 工作流序列后，我将创建所需文件。步骤 1–7 是液体处理操作，应归入单个 OT-2 协议；步骤 8–12 则涉及 YAML 工作流中的其他模块。

文件 1：pcr_reaction_setup.ot2_protocol

```
    '''python requirements = {"robotType":"
OT-2"} from opentrons import 实验方案_api

metadata = { "protocolName": "PCR 反应配置", "author": "PRISM", "description": "使用主混合液、引物、模板DNA和对
照完成 PCR 反应组装", "apiLevel": "2.12", "info": "自动化 PCR 反应配置实验方案", "name": "PCR 反应配置", "version": "
1.0" }

def run(protocol: protocol_api.协议上下文): 实验台 =
{} 移液器 = {}

    ################ # 加载实验室器具 # ################ # 在位置 3 的温控模块上加载反应
    _微孔板 module = protocol.load_module("温控模块", "3") deck["3"] = module.load_
    labware("nest_96_wellplate_100ul_pcr_full_skirt") deck["3"].set_offset(x=1.0, y=1.4, z=5.7)
    # 在位置 1 加载目的微孔板 deck["1"] =protocol.load_labware("nest_96_wellplate_100ul_
    pcr_full_skirt", "1") deck["1"].set_offset(x=0.9, y=-0.5, z=0.6) # 加载吸头架 deck["7"] =
    protocol.load_labware("opentrons_96_tiprack_20ul", "7") deck["7"].set_offset(x=0.2, y=1.
    6, z=-0.8) pipettes["left"] = protocol.load_instrument( "p20_single_gen2", "left", tip_
    racks=[deck["7"]])

    ################### # 执
    行命令 # ################
    ###

    # 步骤 1：转移 20 uL qPCR预混液 # A1→B2 移液器["left"].
    pick_up_吸头() 移液器["left"].孔_bottom_clearance.吸液
    = 1移液器["left"].吸液(20.0, 实验台["3"] ["A1"]) 移液器["
    left"].孔_bottom_clearance.分液 = 1移液器["left"].分液
    (20.0, 实验台["1"] ["B2"]) 移液器["left"].混匀(3, 20,实验台["
    1"] ["B2"]) 移液器["left"].blow_out() 移液器["left"].drop_吸
    头() # A2→B11 移液器["left"].pick_up_吸头() 移液器["left"].
    孔_bottom_clearance.吸液 = 1
```

移液器["left"].吸液(20.0, 实验台["3"]["A2"]) 移液器["left"].孔_bottom_clearance.分液 = 1移液器["left"].分液(20.0, 实验台["1"]["B11"]) 移液器["left"].混匀(3, 20, 实验台["1"]["B11"]) 移液器["left"].blow_out() 移液器["left"].drop_吸头()

A1→G2

移液器["左"].拿_起_吸头() 移液器["左"].孔_底部_间隙.吸液 = 1移液器["左"].吸液(20.0, 实验台["3"]["A1"]) 移液器["左"].孔_底部_间隙.分液 = 1移液器["左"].分液(20.0, 实验台["1"]["G2"]) 移液器["左"].混匀(3, 20, 实验台["1"]["G2"]) 移液器["左"].吹_出() 移液器["左"].丢弃_吸头()

A2→G11

移液器["左"].拿_起_吸头() 移液器["左"].孔_底部_间隙.吸液 = 1移液器["左"].吸液(20.0, 实验台["3"]["A2"]) 移液器["左"].孔_底部_间隙.分液 = 1移液器["左"].分液(20.0, 实验台["1"]["G11"]) 移液器["左"].混匀(3, 20, 实验台["1"]["G11"]) 移液器["左"].吹_出() 移液器["左"].丢弃_吸头()

步骤 2：转移 5 uL 无核酸酶生物水 # C1→B2

移液器["left"].拾_取_吸头() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"]["C1"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"]["B2"]) 移液器["left"].混匀(3, 20, 实验台["1"]["B2"]) 移液器["left"].吹_出() 移液器["left"].丢_弃吸头()

C2→B11

移液器["left"].拾_取_吸头() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"]["C2"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"]["B11"]) 移液器["left"].混匀(3, 20, 实验台["1"]["B11"]) 移液器["left"].吹_出() 移液器["left"].丢_弃吸头()

C3→G2

移液器["left"].拾取_上_吸头() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"]["C3"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"]["G2"]) 移液器["left"].混匀(3, 20, 实验台["1"]["G2"]) 移液器["left"].吹出() 移液器["left"].丢弃_吸头()

C4→G11

移液器["left"]。拾取_吸头_() 移液器["left"]。孔_底部_间隙。吸液 = 1移液器["left"]。吸液(5.0, 实验台["3"]["C4"]) 移液器["left"]。孔_底部_间隙。分液 = 1移液器["left"]。分液(5.0, 实验台["1"]["G11"]) 移液器["left"]。混匀(3, 20, 实验台["1"]["G11"]) 移液器["left"]。吹_出() 移液器["left"]。丢弃_吸头()

步骤 3：将 5 uL 正向引物从 D1 转移至所有目标位置 # D1→B2 移液器["left"]。拾取_吸头_() 移液器["left"]。孔_底部_间隙。吸液 = 1移液器["left"]。吸液(5.0, 实验台["3"]["D1"]) 移液器["left"]。孔_底部_间隙。分液 = 1移液器["left"]。分液(5.0, 实验台["1"]["B2"]) 移液器["left"]。混匀(3, 20, 实验台["1"]["B2"]) 移液器["left"]。吹_出() 移液器["left"]。丢弃_吸头()

D1→B11

移液器["left"].拾_取_吸头() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"]["D1"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"]["B11"]) 移液器["left"].混匀(3, 20, 实验台["1"]["B11"]) 移液器["left"].吹_出() 移液器["left"].丢_弃吸头()

D1→G2

移液器["left"].拾_取_吸头() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"]["D1"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"]["G2"]) 移液器["left"].混匀(3, 20, 实验台["1"]["G2"]) 移液器["left"].吹_出() 移液器["left"].丢_弃吸头()

D1→G11 移液器["left"].pick_up_tip() 移液器["left"].well_bottom_clearance.吸液 = 1移液器["left"].aspirate(5.0, 实验台["3"]["D1"]) 移液器["left"].well_bottom_clearance.分液 = 1移液器["left"].dispense(5.0, 实验台["1"]["G11"]) 移液器["left"].mix(3, 20, 实验台["1"]["G11"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

步骤 4：将 5 uL 反向引物从 E1 转移到所有目标位置 # E1→B2 移液器["left"].pick_up_tip() 移液器["left"].well_bottom_clearance.吸液 = 1移液器["left"].aspirate(5.0, 实验台["3"]["E1"]) 移液器["left"].well_bottom_clearance.分液 = 1移液器["left"].dispense(5.0, 实验台["1"]["B2"]) 移液器["left"].mix(3, 20, 实验台["1"]["B2"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

E1→B11

移液器["左侧"].取_出_吸头() 移液器["左侧"].孔_底部_间隙.吸液 = 1移液器["左侧"].吸液(5.0, 实验台["3"]["E1"]) 移液器["左侧"].孔_底部_间隙.分液 = 1移液器["左侧"].分液(5.0, 实验台["1"]["B11"]) 移液器["左侧"].混匀(3, 20, 实验台["1"]["B11"]) 移液器["左侧"].吹_出() 移液器["左侧"].丢_弃吸头()

E1→G2

移液器["左侧"].拾取_吸头_() 移液器["左侧"].孔_底部_间隙.吸液 = 1移液器["左侧"].吸液(5.0, 实验台["3"]["E1"])

移液器["左侧"].孔_底部_间隙.分液 = 1移液器["左侧"].分液(5.0, 实验台["1"] ["G2"]) 移液器["左侧"].混匀(3, 20, 实验台["1"] ["G2"]) 移液器["左侧"].吹出_() 移液器["左侧"].丢弃_吸头()

E1→G11 移液器["left"].pick_up_tip() 移液器["left"].孔_底部_间隙.aspirate = 1移液器["left"].吸液(5.0, 实验台["3"] ["E1"]) 移液器["left"].孔_底部_间隙.dispense = 1移液器["left"].分液(5.0, 实验台["1"] ["G11"]) 移液器["left"].混匀(3, 20, 实验台["1"] ["G11"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

步骤 5：将 5 uL 模板DNA从 H1 转移至样品孔（不包括对照） # H1→B2 移液器["left"].pick_up_tip() 移液器["left"].孔_底部_间隙.aspirate = 1移液器["left"].吸液(5.0, 实验台["3"] ["H1"]) 移液器["left"].孔_底部_间隙.dispense = 1移液器["left"].分液(5.0, 实验台["1"] ["B2"]) 移液器["left"].混匀(3, 20, 实验台["1"] ["B2"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

H1→G2 移液器["左侧"].吸取_上_吸头() 移液器["左侧"].孔_底部_间隙.吸液 = 1移液器["左侧"].吸液(5.0, 实验台["3"] ["H1"]) 移液器["左侧"].孔_底部_间隙.分液 = 1移液器["左侧"].分液(5.0, 实验台["1"] ["G2"]) 移液器["左侧"].混匀(3, 20, 实验台["1"] ["G2"]) 移液器["左侧"].吹_出() 移液器["左侧"].丢弃_吸头()

步骤 6：将 5 uL 无核酸酶生物水转移至对照孔（而非模板） # C2→B11 移液器["左侧"].吸取_上_吸头() 移液器["左侧"].孔_底部_间隙.吸液 = 1移液器["左侧"].吸液(5.0, 实验台["3"] ["C2"]) 移液器["左侧"].孔_底部_间隙.分液 = 1移液器["左侧"].分液(5.0, 实验台["1"] ["B11"]) 移液器["左侧"].混匀(3, 20, 实验台["1"] ["B11"]) 移液器["左侧"].吹_出() 移液器["左侧"].丢弃_吸头()

C4→G11 移液器["left"].pick_up_tip() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].吸液(5.0, 实验台["3"] ["C4"]) 移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].分液(5.0, 实验台["1"] ["G11"]) 移液器["left"].混匀(3, 20, 实验台["1"] ["G11"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

第 7 步：在所有目标孔中以 20uL 混匀 10 个循环 # 混匀 B2 移液器["left"].pick_up_tip() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].混匀(10, 20,实验台["1"] ["B2"]) 移液器["left"].吹_出() 移液器["left"].丢弃_吸头()

混匀 G2 移液器["left"].pick_up_tip() 移液器["left"].孔_底部_间隙.吸液 = 1移液器["left"].孔_底部_间隙.分液 = 1移液器["left"].混匀(10, 20,实验台["1"] ["G2"]) 移液器["left"].吹_出() 移液器["left"].丢弃_吸头()

混匀 B11 移液器["left"].pick_up_tip() 移液器["left"].well_bottom_clearance.吸液 = 1移液器["left"].well_bottom_clearance.分液 = 1移液器["left"].混匀(10, 20,实验台["1"] ["B11"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

混匀 G11 移液器["left"].pick_up_tip() 移液器["left"].well_bottom_clearance.吸液 = 1移液器["left"].well_bottom_clearance.分液 = 1移液器["left"].混匀(10, 20,实验台["1"] ["G11"]) 移液器["left"].blow_out() 移液器["left"].drop_tip()

‘‘‘

文件 2：pcr_工作流程.yaml

“‘yaml 名称：自动化 PCR 工作流程 作者：PRISM

信息：包含反应配置、热循环和板读取的完整自动化 PCR 工作流程 版本：’1.0’

flowdef: ########################## # 液体处理设置# ########################## - 操作: run_规程名称: 执行 PCR 反应配置协议模块: ot2bioalpha 文件: 实验方案: payload.pcr_反应_配置.ot2_实验方案

################################## # 转移至封板机（通过交换站） ################################## - 操作: 转移 名称: 将微孔板从 OT-2 转移至交换站 模块: biopf400 参数: 源: ot2bioalpha_实验台1_宽 源_接近方式: 安全_路径_ot2bioalpha 源_微孔板_旋转: 宽 目标: 交换_实验台_高_宽 目标_接近方式: 安全_路径_交换 目标_微孔板_旋转: 宽

- 操作：转移 名称：将微孔板从交换站转移至封板机并进行方向变更 模块：biopf400 参数：源：交换站_实验台_高_窄 源_接近方式：安全_路径_交换站 源_微孔板_旋转：窄 目标：封板机_托架 目标_接近方式：安全_路径_封板机 目标_微孔板_旋转：窄

########## # 封板 ########## - 操作:
封板 名称：封板聚合酶链式反应反应微孔
板 模块：bio_封板机 参数：{}

################################### # 转移至热循环仪（经由交
换站） ################################### - 操作: 转移 名称: 将
封板后的微孔板从封板机转移至交换站 模块: biopf400 参数: 源: sealer_
nest source_approach: safe_path_sealer source_plate_rotation: 窄 目
标: exchange_deck_high_narrow target_approach: safe_path_
exchange target_plate_rotation: 窄

- 操作: 转移 名称: 将微孔板从交换站转移至热循环仪并改变方向 模块: biopf400 参数: 源:
exchange_deck_high_wide source_approach: safe_path_exchange source_plate_
rotation: 宽 目标: bio_biometra3_nest target_approach: safe_path_biometra3 target_
plate_rotation: 宽

############### # 热
循环 ##############
#
- 操作: 打开 名称: 打开热循环仪盖
子 模块: bio_biometra3_96参数:
{}

- 操作: 关闭 名称: 关闭热循环仪盖
子 模块: bio_biometra3_96参数:
{}

- 操作: 运行_程序名称: 执行聚合酶链式反应热循
环程序 模块: bio_biometra3_96参数: 程序_编号:
5

- 操作: 打开 名称: 完成后打开热循环仪盖 模块: bio_
biometra3_96参数: {}

############################### # 转移至撕膜机
（通过交换站） ###############################
#
- 操作：转移 名称：从热循环仪转移微孔板到交换站 模块：biopf400 参
数：源：bio_biometra3_托架源_接近方式：safe_路径_biometra3

源_微孔板_旋转：宽 目标：交换_实验台_
高_宽 目标_接近方式：safe_路径_交换 目
标_微孔板_旋转：宽

- 操作：转移 名称：将微孔板从交换站转移至撕膜机（方向变更）模块：biopf400
参数：源：交换站_实验台_高_窄 源_接近方式：安全_路径_交换站 源_微孔板_旋转：
窄 目标：撕膜机_托架 目标_接近方式：安全_路径_撕膜机 目标_微孔板_旋转：窄

########## # 剥离 ########## - 操作：
剥离 名称：从聚合酶链式反应微孔板上
移除封膜模块：bio_撕膜机 参数：{}

################################### # 转移至板式读数仪（经交
换站） ################################### - 操作：转移 名称：
将已剥离封膜的微孔板从撕膜机转移至交换站 模块：biopf400 参数：源：
撕膜机_托架 源_接近方式：安全_路径_撕膜机 源_微孔板_旋转：窄 目标：
交换站_实验台_高_窄 目标_接近方式：安全_路径_交换站 目标_微孔板_旋
转：窄

- 操作：转移 名称：将微孔板从交换站转移至Hidex酶标仪 模块：
biopf400 参数：源：交换站_实验台_高_窄 源_接近方式：安全_路径_
交换站 源_微孔板_旋转：窄 目标：hidex_geraldine_高_托架 目标_接
近方式：安全_路径_hidex 目标_微孔板_旋转：窄

############### # 板读取 ##########
- 操作：打开 名称：打开 Hidex 读
板仪盖 模块：hidex_geraldine 参数：{}

- 操作：关闭 名称：关闭 Hidex 酶标仪盖
模块：hidex_geraldine 参数：{}

- 操作：运行_检测 名称：运行聚合酶链式
反应最终结果检测 模块：hidex_
geraldine 参数：检测_名称："聚合酶链
式反应_最终_结果"

- 操作：打开 名称：分析后打开 Hidex酶标仪盖 模块：
hidex_geraldine 参数：{} “‘

这些文件实现了完整的PCR工作流程：

pcr_反应_设置.ot2_实验方案：将所有液体处理步骤（1-7）整合到一个OT-2实验方案中，该方案：
- 向所有目标孔中加入qPCR预混液 - 加入无核酸酶水 - 向所有孔中加入正向和反向引物 - 向样品孔（B2、G2）中加入模板DNA - 在对照孔（B11、G11）中加入水而非模板 - 对所有孔进行最终混匀

聚合酶链式反应_工作流程.yaml：编排整个工作流程，包括：
- 执行液体处理协议 - 使用交换站在模块之间转移微孔板，以进行方向变更 - 执行封板、热循环、剥离和板读取操作 - 遵循所有模块限制（打开/关闭盖子、确保正确的板方向）

B.3 带仿真反馈提示的迭代

以下是根据上一节输出的仿真结果提供给 Claude Sonnet4.5 模型的完整响应，该响应提示 Claude 识别并修复错误。此过程会针对每个新错误重复进行，直至仿真检测不到任何错误。

执行步骤 6 时出错（‘操作=转移，模块=biopf400，参数={’源’: ’交换站_实验台_高_宽’, ’源_接近方式’: ’安全_路径_交换站’, ’源_微孔板_旋转’: ’宽’, ’目标’: ’bio_biometra3_ 托架 ’, ’ 目标 _ 接近方式 ’: ’ 安全 _ 路径 _biometra3’,’ 目标 _ 微孔板 _ 旋转 ’: ’ 宽 ’}‘ ）： 微孔板与 bio_biometra3_96 发生碰撞。请应用更正后重写整个 YAML文件。