



Institute of Particle Physics
粒子物理研究所

CollierAgent

袁兴博
华中师范大学

文章和作者

arXiv: 2603.14553

RISE-AGI-2026-001
CPTNP-2026-012

An End-to-end Architecture for Collider Physics and Beyond

仇是 蔡则宇 卫家燊 李泽宇 殷艺轩
Shi Qiu,¹ Zeyu Cai,¹ Jiashen Wei,¹ Zeyu Li,^{1,2} Yixuan Yin,¹ Qing-Hong
Cao,^{1,3,2} Chang Liu,^{1,4} Ming-xing Luo,⁵ Xing-Bo Yuan,⁶ and Hua Xing Zhu^{1,2}
曹庆宏 刘畅 罗民兴 袁兴博 朱华星

¹*School of Physics, Peking University, Beijing 100871, China*

²*Center for High Energy Physics, Peking University, Beijing 100871, China*

³*School of Physics, Zhengzhou University, Zhengzhou 450001, China*

⁴*State Key Laboratory of Nuclear Physics and Technology, Peking University, Beijing 1008*

⁵*Beijing Computational Science Research Center, Beijing 100193, China*

⁶*Institute of Particle Physics and Key Laboratory of Quark and Lepton Physics (MOE), Central China Normal University, Wuhan, Hubei 430079, China*

群聊: ColliderAgent 交流群



项目地址: <https://github.com/HET-AGI/ColliderAgent> (安装方法, 使用说明)

该二维码7天内(7月7日前)有效, 重新进入将更新

目录

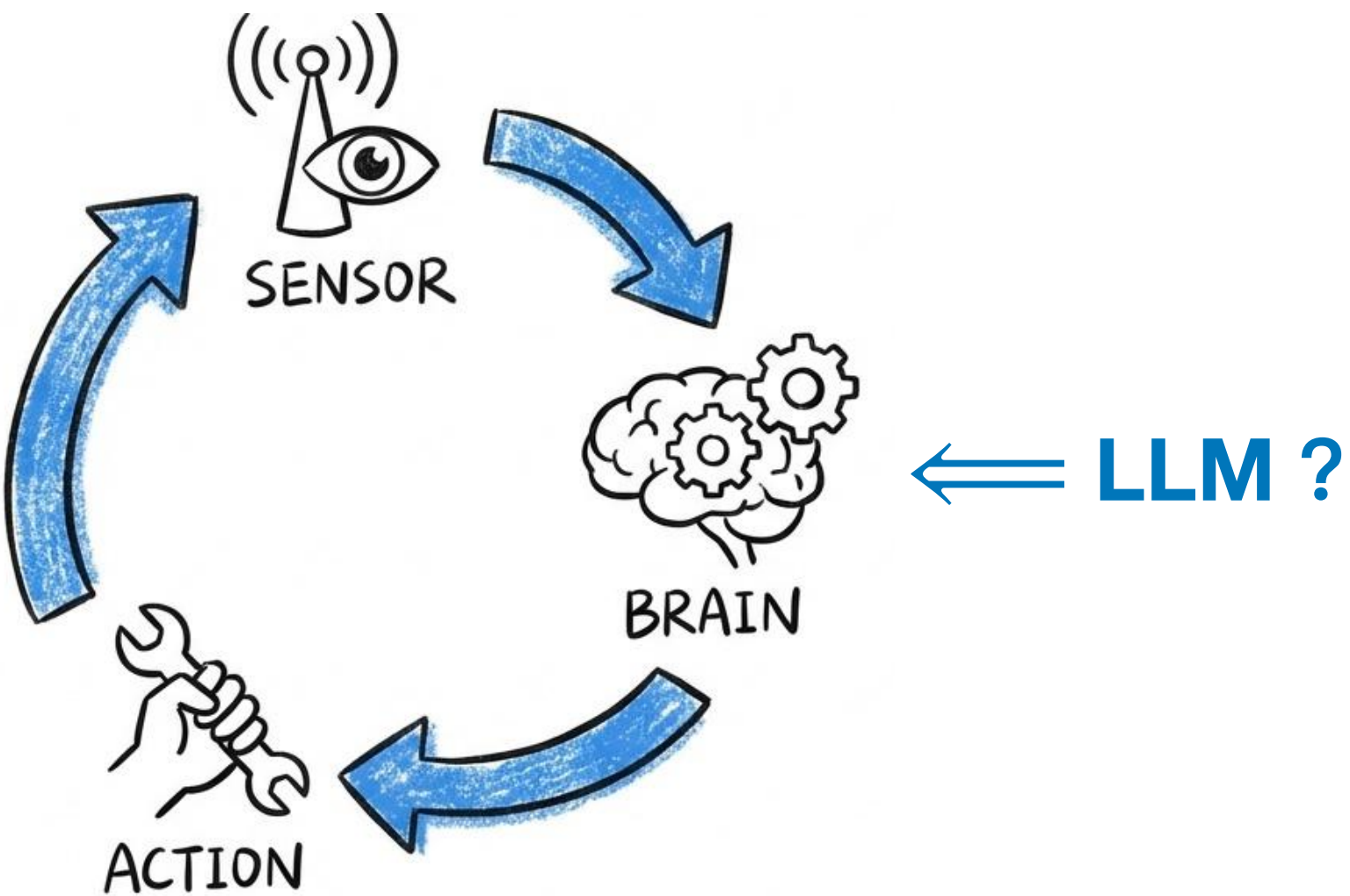
AI Agent

ColliderAgent及其架构

ColliderAgent可以做什么？

总结和展望

AI Agent

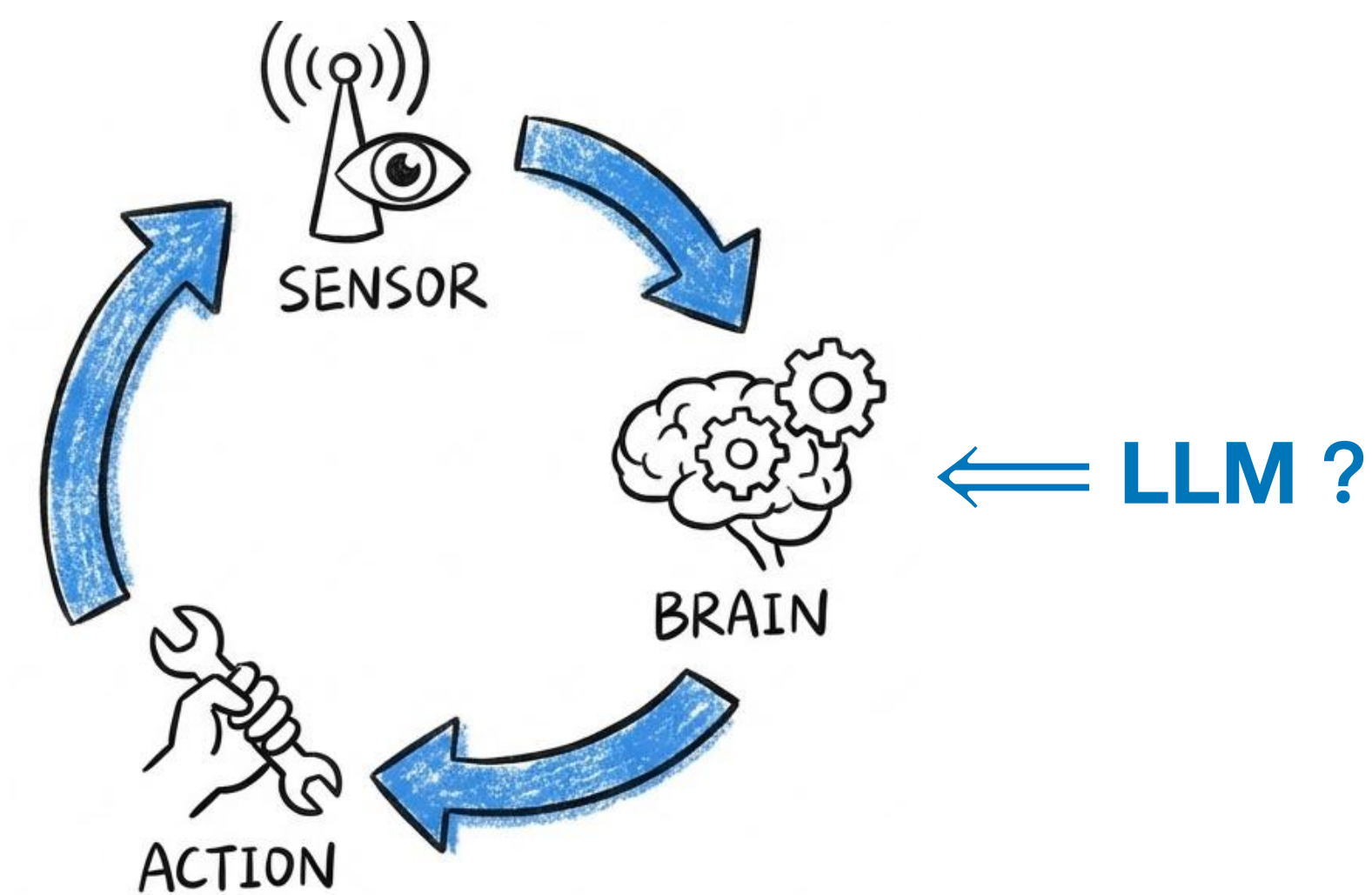
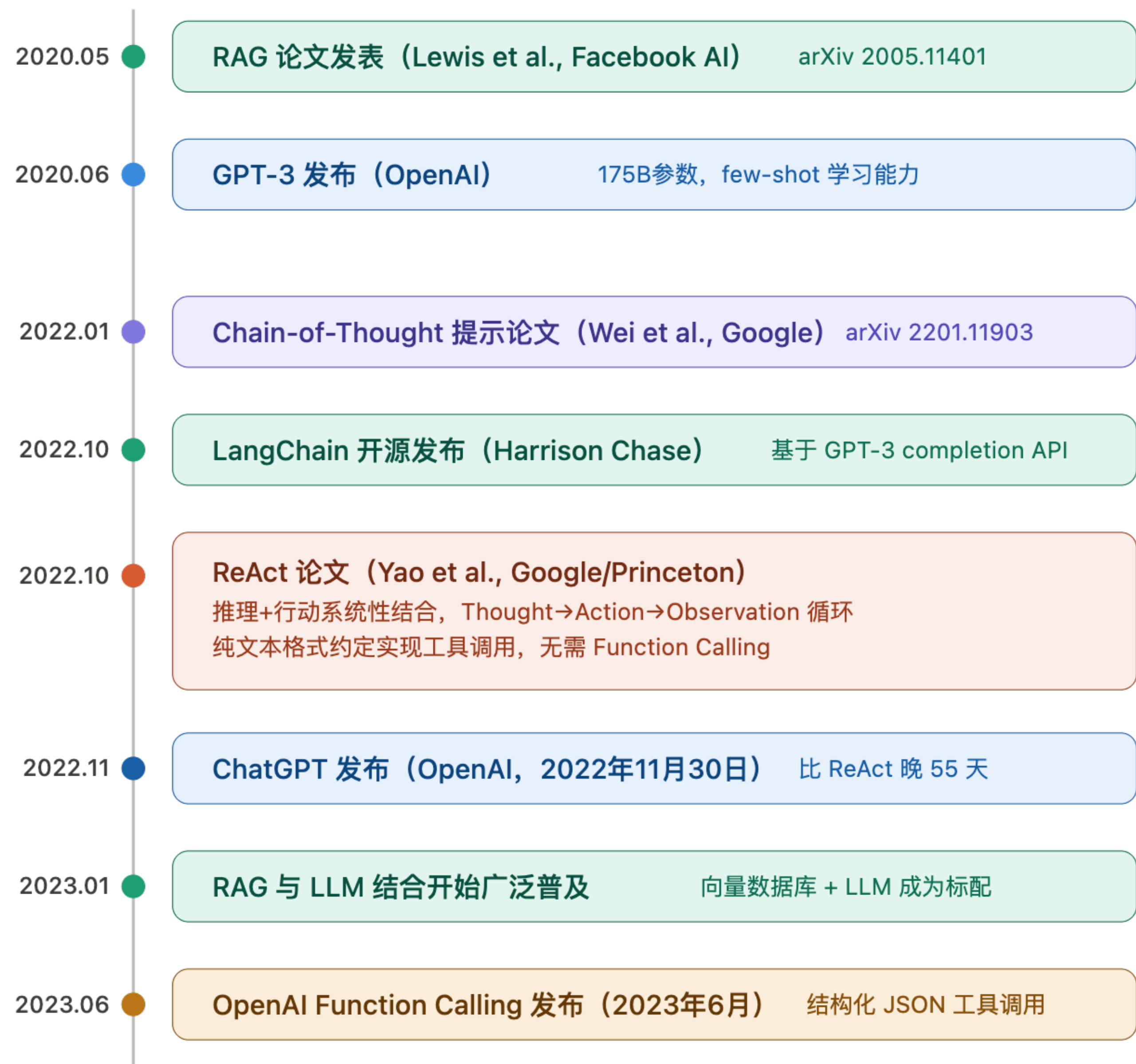


Russell & Norvig, *AIMA*: Agent 是任何能够感知环境（使用传感器）并对其做出反应（使用执行器）的物体。

例如：带有摄像头和轮子的机器人，或者读取数据并提出建议的软件程序。

在AI时代，能否用LLM作为Agent的大脑？

AI Agent



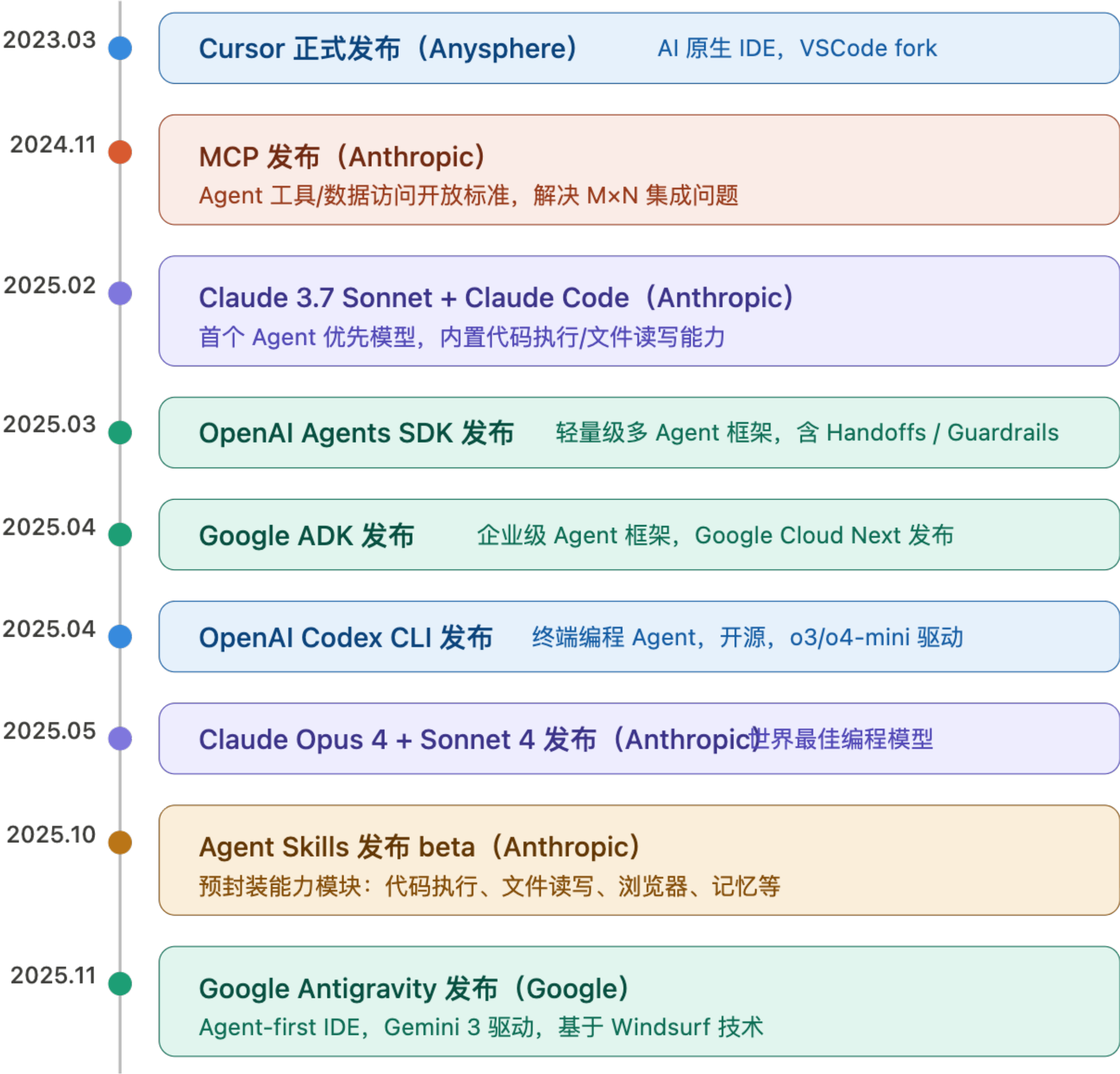
能否用LLM作为Agent的大脑？

不行（2020年，GPT-3发布）

- ▶ LLM缺乏推理能力：不知道要用什么工具来完成任务
- ▶ LLM没有工具调用能力：不会使用工具

但是，2022年被掩盖在ChatGPT光辉下的两项研究改变了这一事实：**CoT和ReAct**

AI Agent

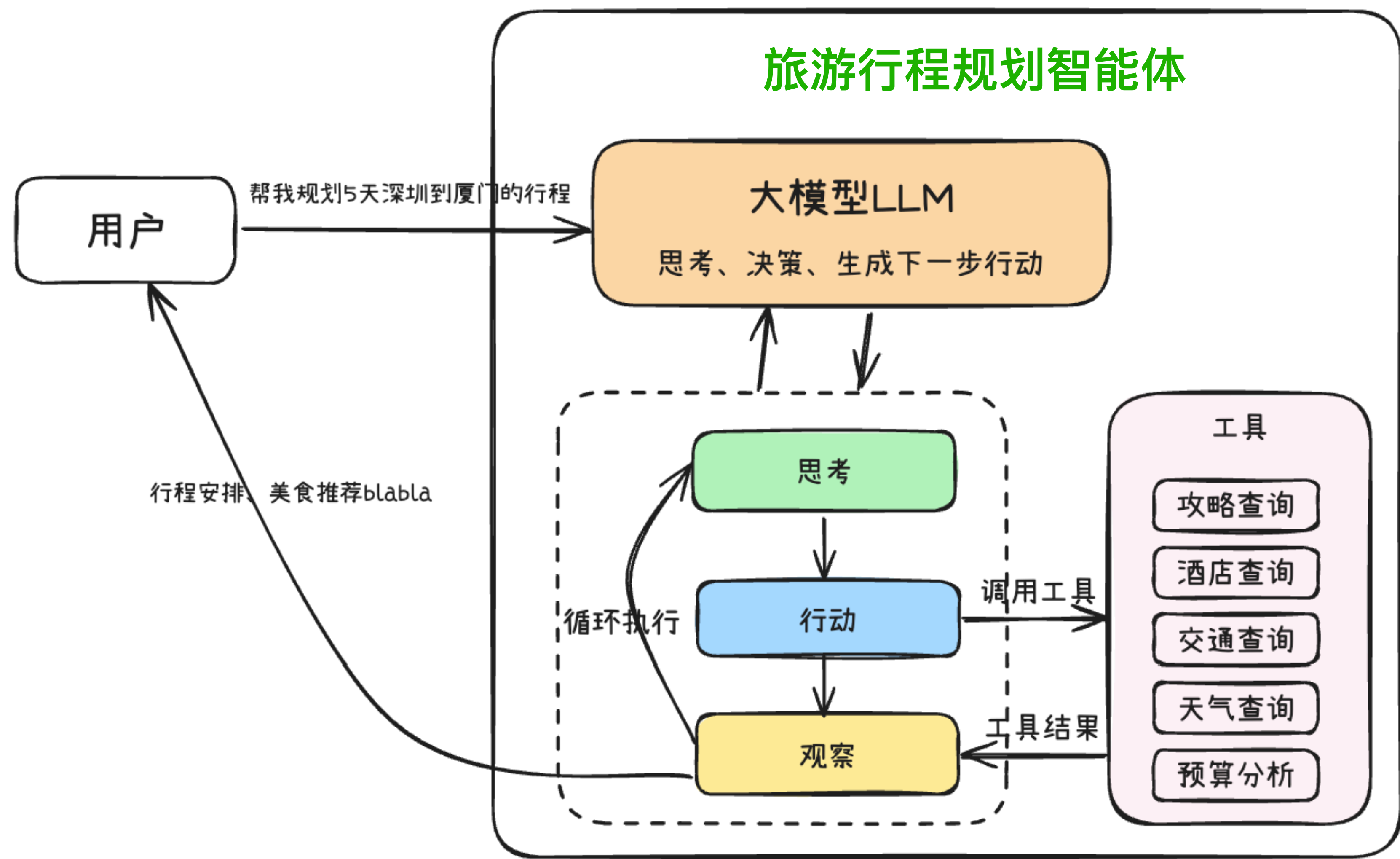


模型能力进一步提升：**Claude Opus 4**

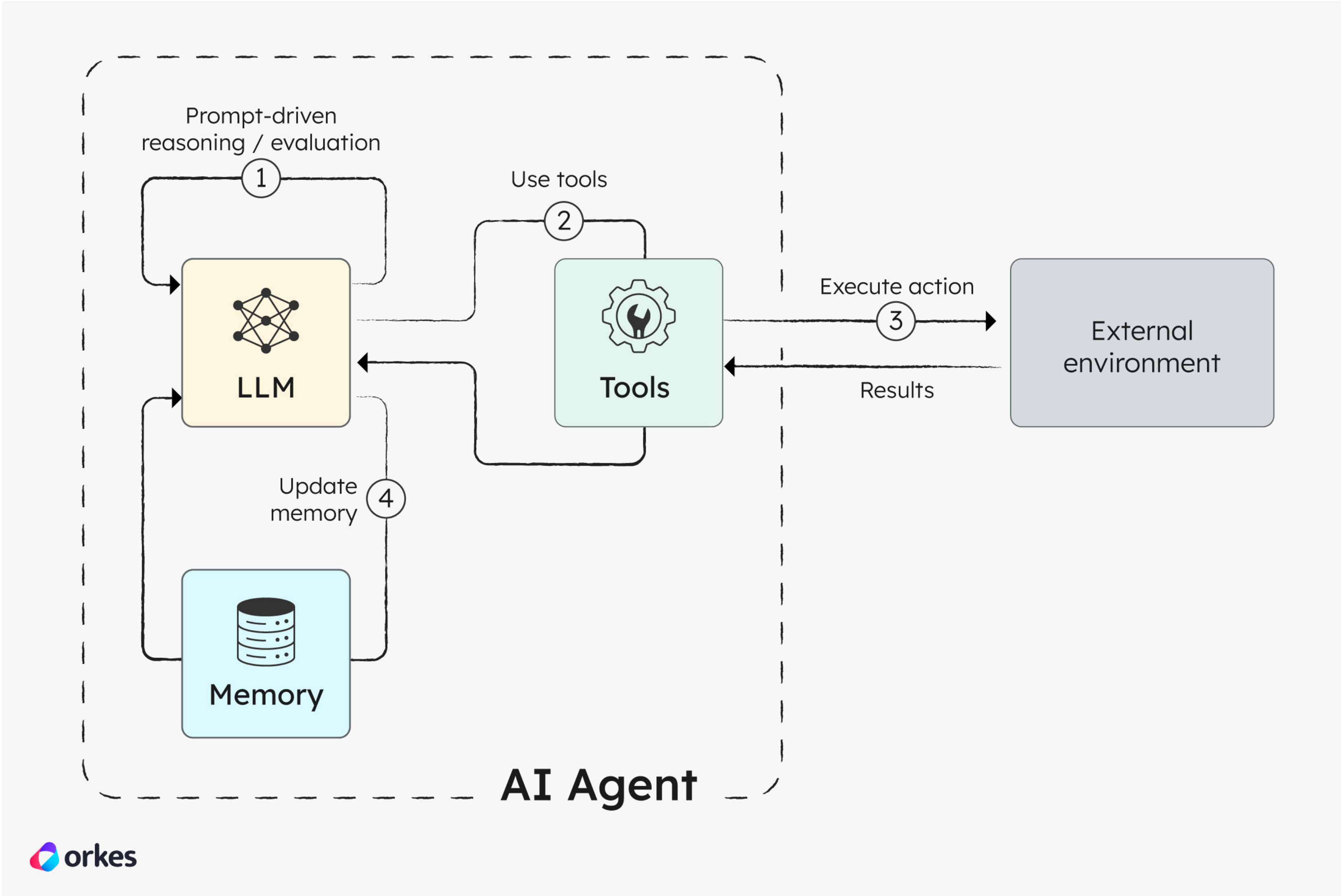
工具调用进一步规范：**MCP, Agent Skills**

得益于这些进展，**AI Agent**开始从实验室逐渐向生产力转变，**2025年**也被称为“**智能体元年**”

典型的AI智能体



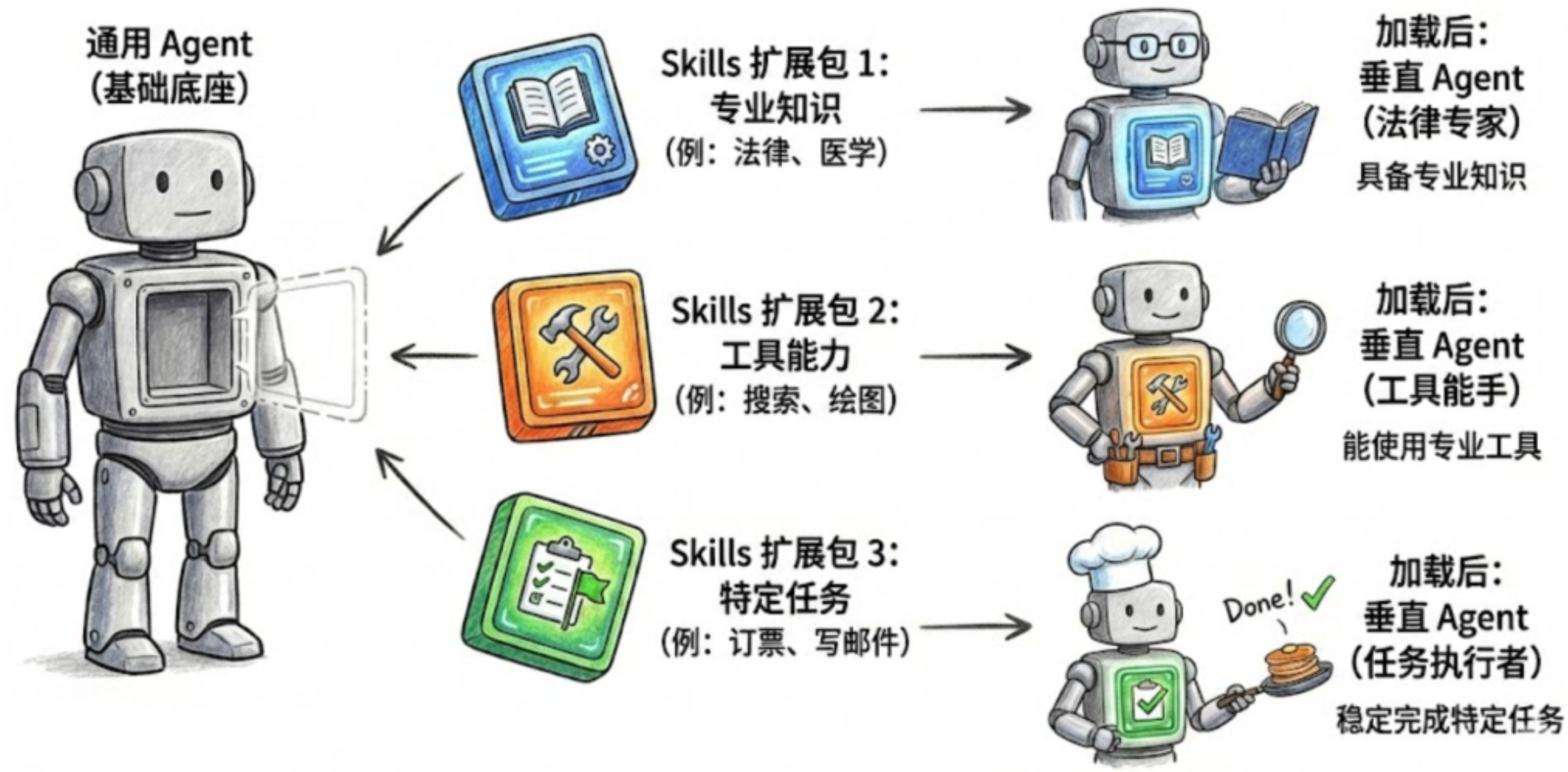
典型的AI智能体



智能体相关技术： Skills

Agent Skills概念是由 Anthropic 公司于 2025 年10月提出、12月正式开源的，是一种用于增强智能体在特定任务中专业能力的结构化、模块化知识封装机制。

它的核心目标是：让通用大模型在需要时，按需加载并执行高度专业化的工作流程或领域知识，而不是一次性塞入所有信息。



智能体相关技术： Skills

用于PDF处理的SKILLS.md文件内容

```
---
name: pdf-processing
description: 从 PDF 中提取文本和表格，填写表单，并合并文档
---

# PDF 处理

## 使用场景
当需要对 PDF 文件进行操作时使用，例如：

- 提取 PDF 文本或表格数据
- 填写 PDF 表单
- 合并多个 PDF 文件

## 提取文本
- 使用 `pdfplumber` 提取文本型 PDF 内容
- 扫描版 PDF 需配合 OCR 工具

## 填写表单
- 读取 PDF 表单字段
- 按输入数据填充并生成新文件
```

容易使用： 基于简单的 Markdown 文件

决定agent何时加载。例如：“请帮我提取application.pdf中的表格”时才会触发

渐进加载： 最小化占用上下文空间，Token 使用效率高

知识复用： 可以把经验、最佳实践、工作流程都写进去

智能体相关技术: Skills

Bundling additional content

pdf/SKILL.md

YAML Frontmatter

```
---
name: pdf
description: Comprehensive PDF toolkit for extracting text and
tables, merging/splitting documents, and filling-out forms.
---
```

Markdown

Overview

This guide covers essential PDF processing operations using Python libraries and command-line tools. For advanced features, JavaScript libraries, and detailed examples, see `./reference.md`. If you need to fill out a PDF form, read `./forms.md` and follow its instructions.

Quick Start

```
```python
from pypdf import PdfReader, PdfWriter
```

#### # Read a PDF

```
reader = PdfReader("document.pdf")
print(f"Pages: {len(reader.pages)}")
```

#### # Extract text

```
text = ""
for page in reader.pages:
 text += page.extract_text()
```

pdf/reference.md

### # PDF Processing Advanced Reference

This document contains advanced PDF processing features, detailed examples, and additional libraries not covered in the main skill instructions.

#### ## pypdfium2 Library (Apache/BSD License)

##### ### Overview

pypdfium2 is a Python binding for PDFium (Chromium's PDF library). It's excellent for fast PDF rendering, image generation, and serves as ...

pdf/forms.md

If you need to fill out a PDF form, first check to see if the PDF has fillable form fields. Run this script from this file's directory:

```
`python scripts/check_fillable_fields <file.pdf>`,
and depending on the result go to either the "Fillable
fields" or "Non-fillable fields" and follow those
instructions.
```

#### # Fillable fields

If the PDF has fillable form fields:

- Run this script from this file's directory:

```
`python scripts/extract_form_field_info.py <input.pdf>
<fields.json>`.
...
```

更多的文档、说明、规范和特例等都可以通过引用的形式被渐进式的加载。



# 智能体相关技术: Skills

## Bundling executable scripts

### pdf/forms.md

If you need to fill out a PDF form, first check to see if the PDF has fillable form fields. Run this script from this file's directory: ``python scripts/check_fillable_fields <file.pdf>``, and depending on the result go to either the "Fillable fields" or "Non-fillable fields" and follow those instructions.

# Fillable fields If the PDF has fillable form fields:  
- Run this script from this file's directory:  
``python ./extract_fields.py <input.pdf> <fields.json>``.  
...

### pdf/extract\_fields.py

```
from pypdf import PdfReader

def write_field_info(pdf_path: str, output_path: str):
 """Extract form fields from PDF and store as JSON."""
 reader = PdfReader(pdf_path)
 fields = get_fields(reader)
 with open(output_path, "w") as f:
 json.dump(fields, f)

... omitted ...

if __name__ == "__main__":
 if len(sys.argv) != 3:
 print(f"Usage: python {sys.argv[0]} <pdf_path> <output_json_path>")
 sys.exit(1)
 write_field_info(sys.argv[1], sys.argv[2])
```

支持执行代码，  
极大的增强了灵  
活性和可扩展性

# 智能体相关技术: Skills

## 相对复杂的Skills的文件结构

```
~/.claude/skills/react-component-review/
├── SKILL.md # 核心指令 + 元数据 (建议控制在 400 行内)
├── templates/ # 常用模板 (Claude 按需读取)
│ ├── functional.tsx.md
│ └── class-component.md
├── examples/ # 优秀/反例 (给 Claude 看标准)
│ ├── good.md
│ └── anti-pattern.md
├── references/ # 规范、规则、禁用词表
│ ├── hooks-rules.md
│ └── naming-convention.md
└── scripts/ # 可执行脚本 (需开启 code execution)
 ├── validate-props.py
 └── check-cycle-deps.sh
```

<https://claude.com/blog/equipping-agents-for-the-real-world-with-agent-skills>

<https://platform.claude.com/docs/zh-CN/agents-and-tools/agent-skills/overview>

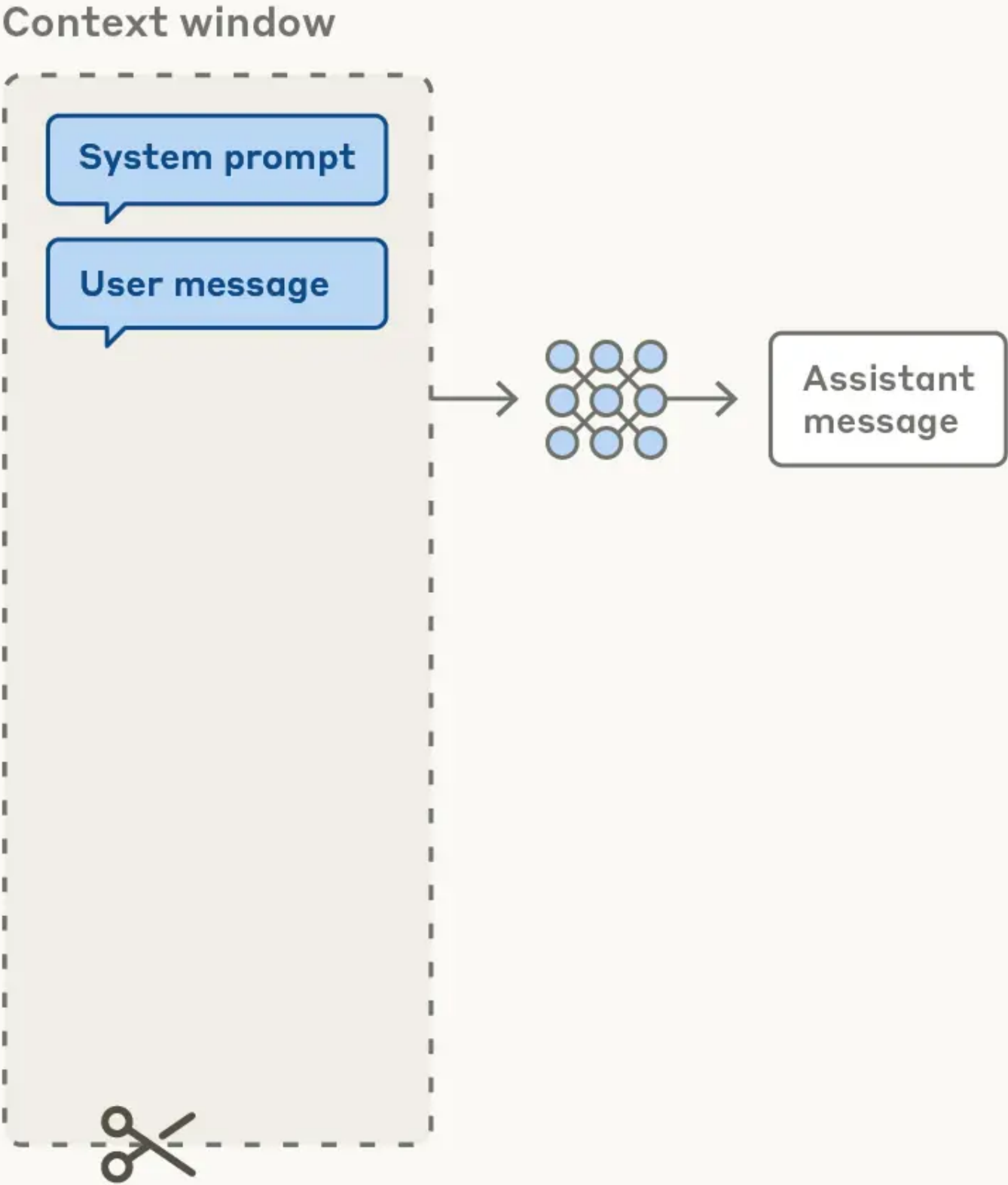


# Context Engineering (上下文工程)

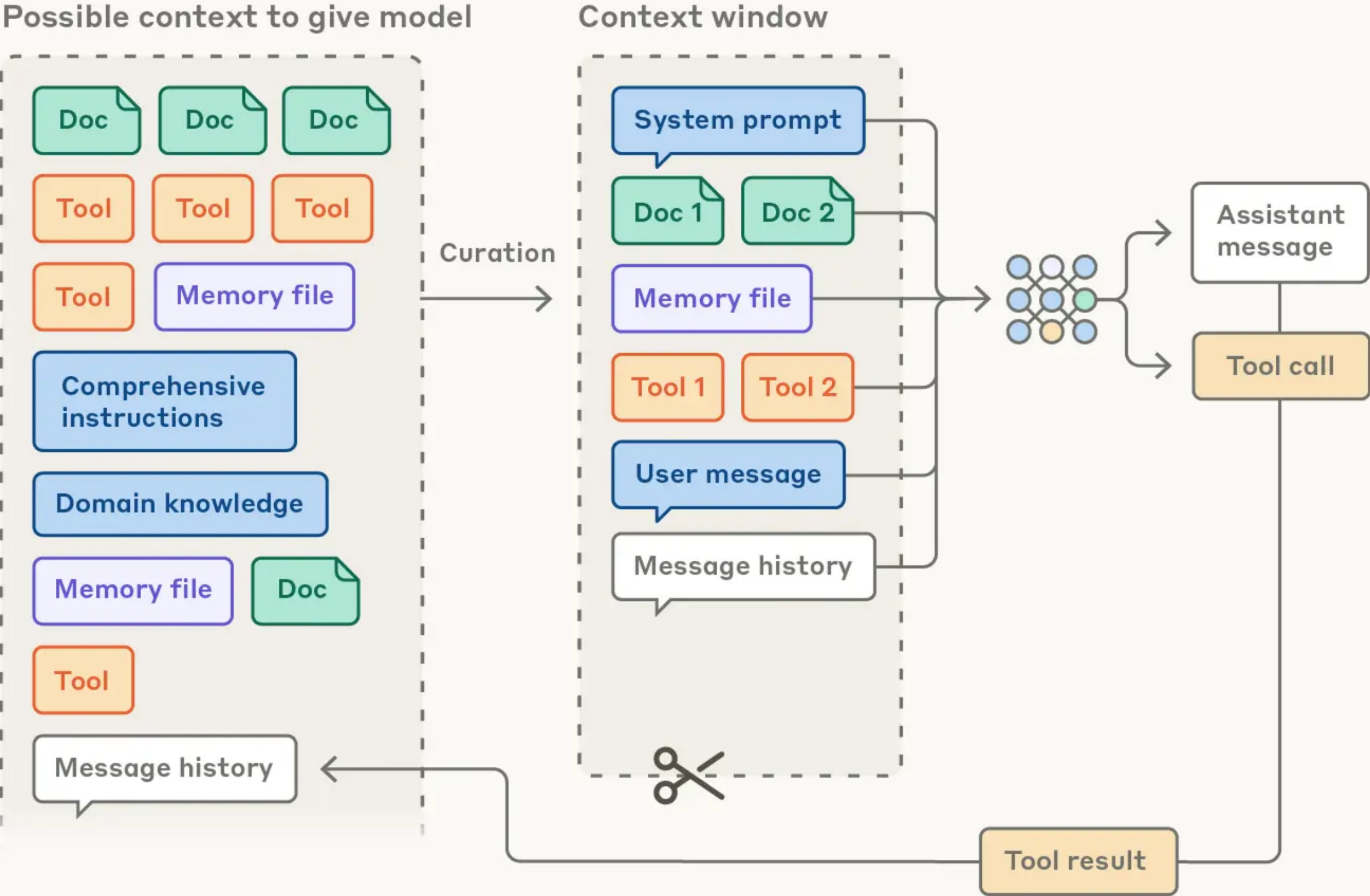
技术上: 上下文工程 ≈ 混合了自然语言的编程

## Prompt engineering vs. context engineering

### Prompt engineering for single turn queries



### Context engineering for agents



上下文工程：  
在与大型语言模型交互时，系统性地设计和管理输入上下文的技术与方法——让模型在每次推理时，拥有完成任务所需的“恰好合适”的信息

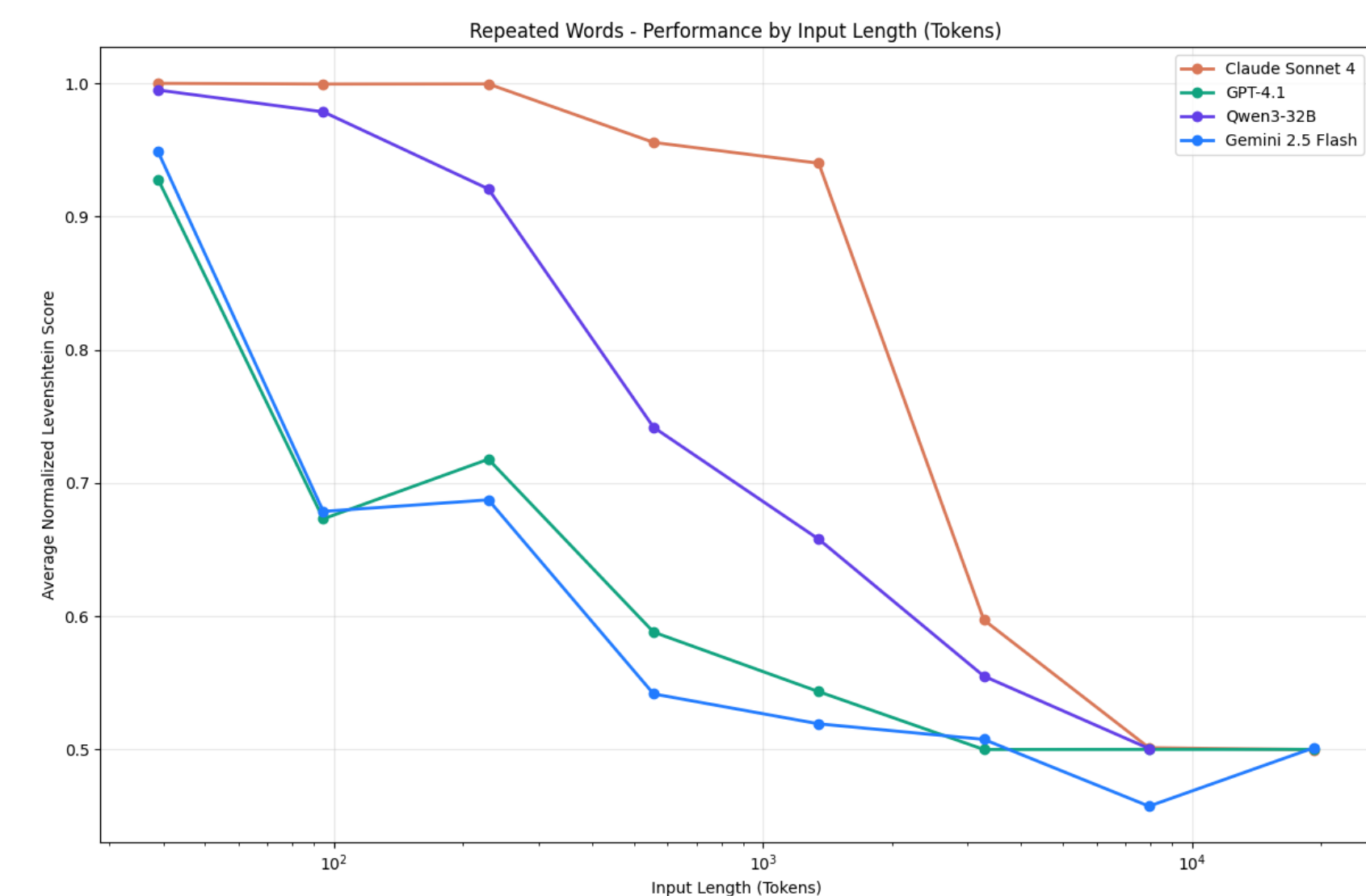
# 上下文空间

上下文空间是智能体的珍贵资源！

上下文腐化：模型的性能会随着输入长度的变化而发生显著变化

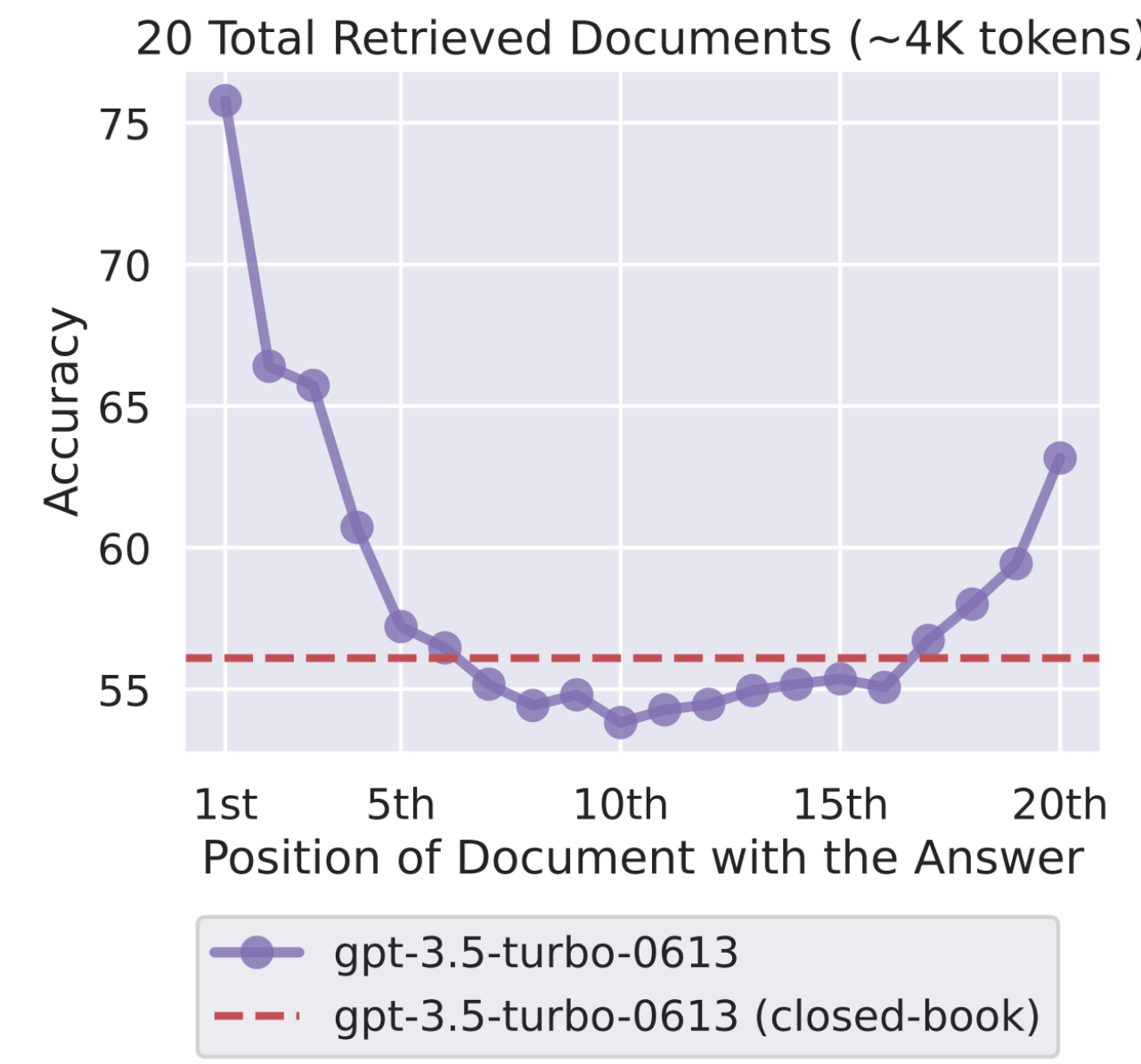
本质原因是当前LLM底层transformer架构的问题，这里不作深入探讨

2025 July <https://www.trychroma.com/research/context-rot>



Lost in the Middle: LLM对长文本信息的注意力是高度不均匀，开头和结尾处更高

N.F. Liu et al, 2307.03172



这些事情告诉我们，为了最大化的利用模型能力，通常只能使用最大上下文的10%~20%

如何在有限的上下文空间中，编排智能体的任务，工具的调用，是agent构建的核心！

特别是，对于复杂的任务，如何充分利用有限的上下文空间是非常具有挑战性的。

U型性能曲线



# 上下文工程与传统编程的区别

	传统编程	上下文工程
输入	精确	模糊
输出	确定	相对确定
长度	无限	有限
测试	单一模块/条件下 只需测试1次	单一模块/条件下 需要测试多次
编写方法	算法，规范	经验/测试
参考资料	丰富的开源程序，如 linux kernel	最好的全部闭源
看起来像	精心编写的代码	普通的文档

- 😄 传统编程要求1个字母都不能输错
- 😓 LLM自身的特性，可通过上下文工程改善
- 😭 Claude Opus 4.6: 一百万上下文  
约等于500页PRD排版的论文，或者1本Peskin QFT
- 😭 多进行非常多的测试，来判断实现效果，烧钱
- 😓 仅有一些经验性的方法（因而是各类Agent的护城河）
- 😓 如Claude Code闭源🙏，缺乏参考资料
- 😓 可能被误认为在写普通文档

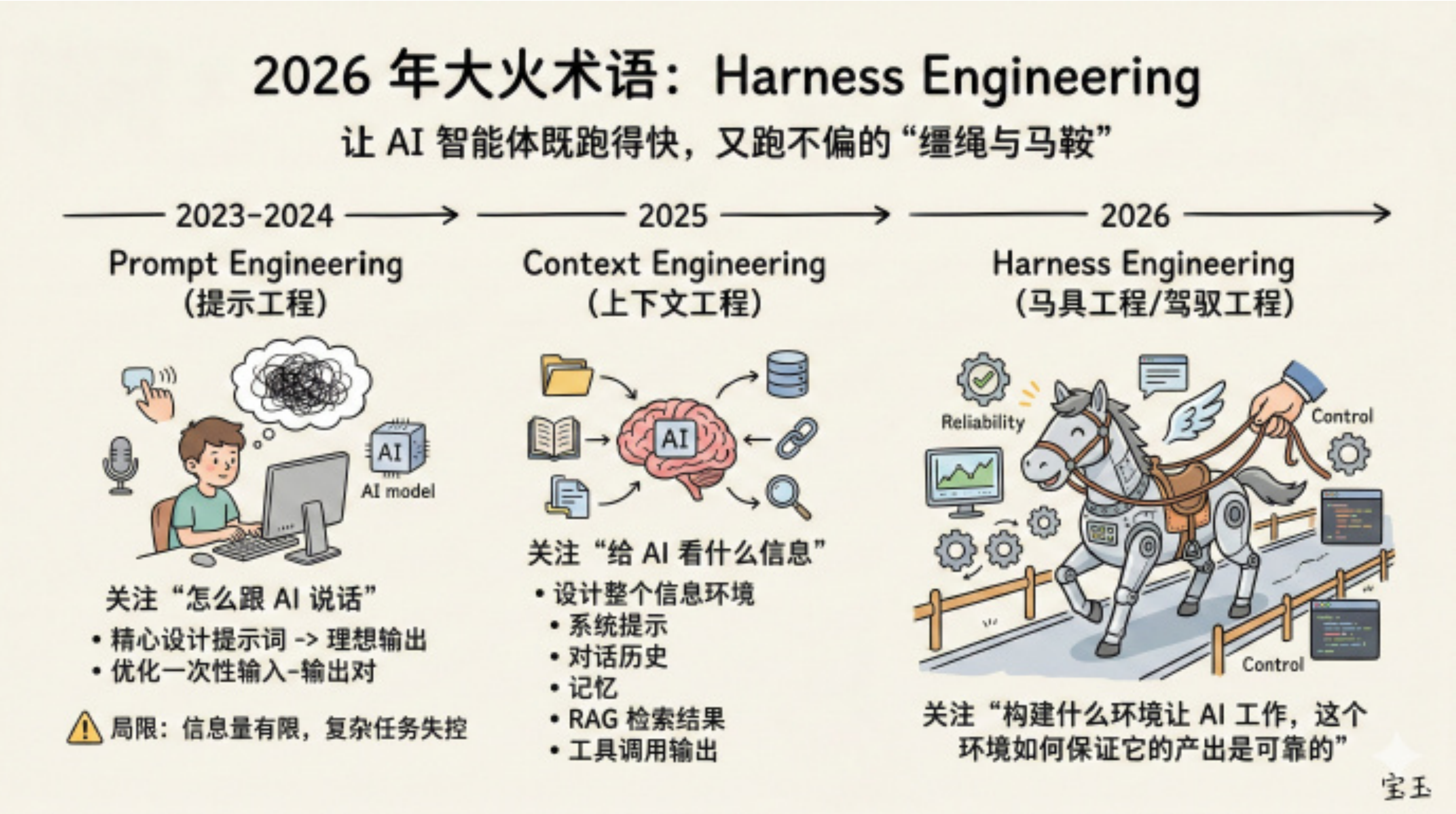


# Harness Engineering

“What changes when a software engineering team’s primary job is no longer to write code, but to design environments, specify intent, and build feedback loops that allow Codex agents to do reliable work.”  
— OpenAI Engineering Blog, *Harness Engineering*

如何让AI在无人监督的情况下，完成复杂的工业级项目

2026.02.11 <https://openai.com/index/harness-engineering/>



当软件工程团队的主要工作不再是写代码,而是设计环境、表达意图,并构建反馈循环,让 AI 智能体能够可靠工作时,软件工程会发生什么变化



# Loop Engineering

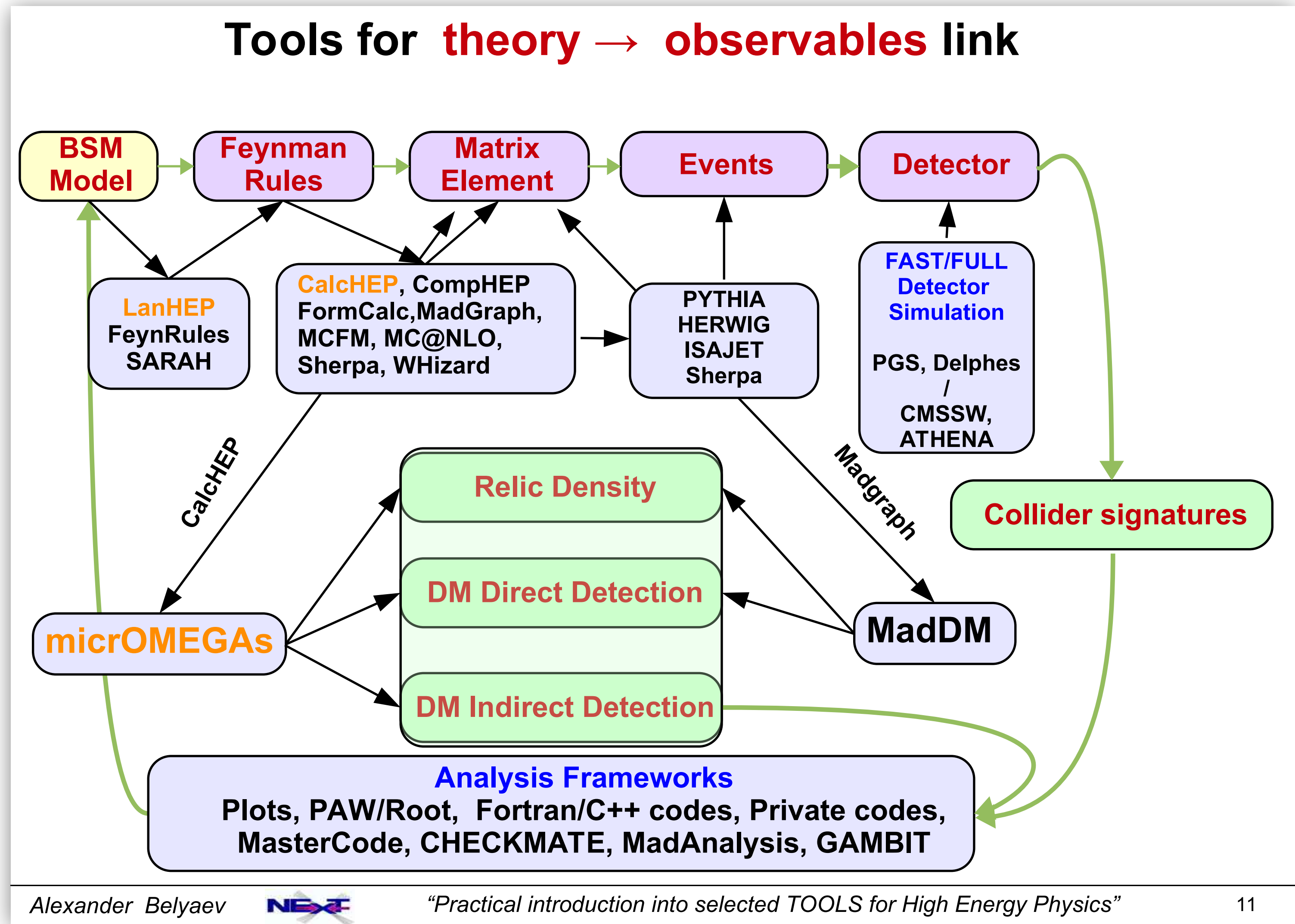
**Addy Osmani: <https://addyosmani.com/blog/loop-engineering/>**



 **核心区别：** 之前是“你推着 AI 走” → 现在是“系统带着 AI 跑”



# 高能物理唯象领域的软件生态



- ▶ 高能物理领域有很多非常优秀的程序包 😊
- ▶ 形成了相对稳定的工作流 😊
- ▶ 不同程序包的代码约定/风格不一样，学起来很难 😞
- ▶ 安装时配置各类环境很烦。使用手册比较粗糙 😞

一线科研人员要花费大量时间学习相关程序包的功能和用法，而要知道程序包内部的运作方式、甚至隐藏的技巧更是难上加难) 😞

能否自动化的运行整个工作流？  
能否减少学习成本？

一些程序尝试自动化工作流的部分阶段、或针对特定唯象任务进行自动化: Checkmate, SARAH/SPheno, GAMBIT GUM

在AI时代，我们能不能做的更好？ 如何去做？

实现工作流的自动化  
降低软件的学习成本  
实现物理结果的可复现性

对撞机物理、暗物质、新物理领域的程序包



# AI与高能物理软件生态的结合

现状：HEP程序包的差异极大，编程语言涵盖C、Fortran、Python、Mathematica，各程序的约定、输入输出的格式也各不相同

问题：如何用AI解决这些问题，并进一步的实现工作流的自动化？

我们发现HEP各类程序包有一个共性：可以通过少量的配置程序，或轻量级的代码来控制

MadGraph  
脚本示例

```
import model sm
define l+ = e+ mu+
define l- = e- mu-
generate p p > l+ l-
output events/pp_ll_14tev
launch events/pp_ll_14tev
done
set nevents 50000
set ebeam1 6500
set ebeam2 6500
```

FeynRules 模型文件示例

```
LALP := Block[{mu, nu, rho, sigma},
- gaZA * (1/2) * Eps[mu, nu, rho, sigma] * FS[Z, mu, nu] * FS[A, rho, sigma] *
alp
- gaZZ * (1/4) * Eps[mu, nu, rho, sigma] * FS[Z, mu, nu] * FS[Z, rho, sigma] *
alp
- gaWW * (1/2) * Eps[mu, nu, rho, sigma] * FS[W, mu, nu] * FS[Wbar, rho, sigma]
* alp
];
```

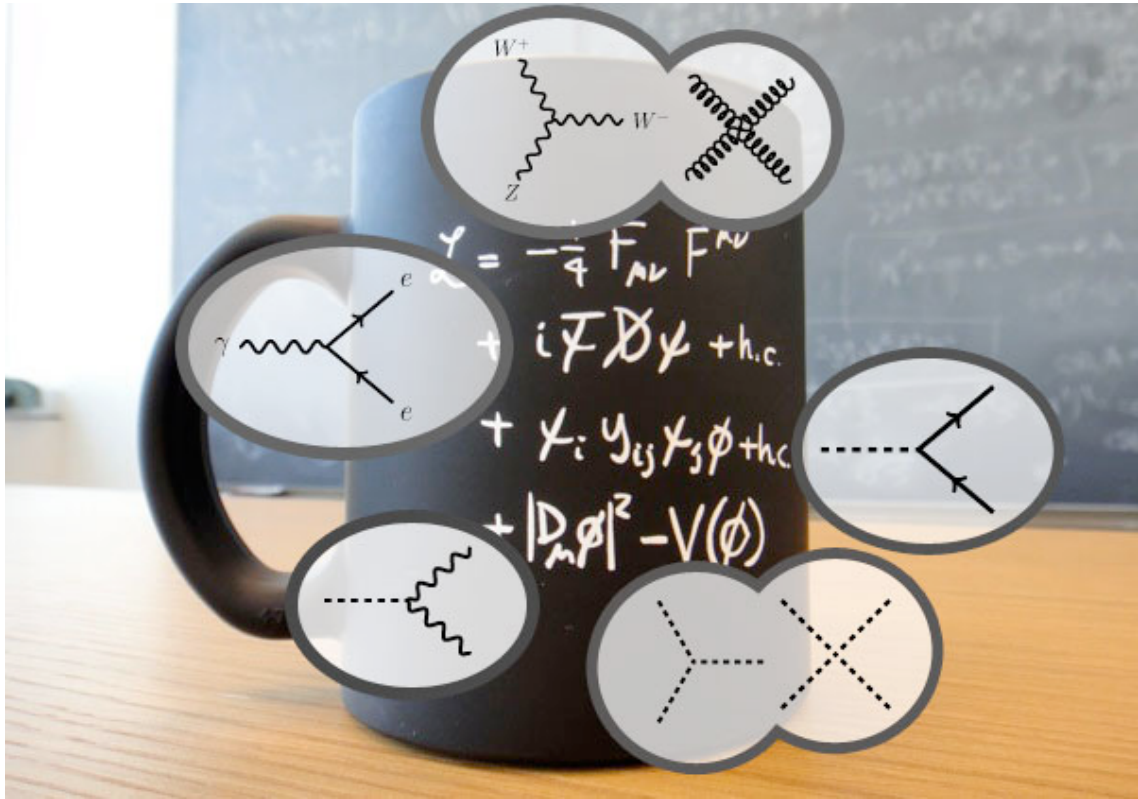
解决方案：如果可以让LLM来写这些配置程序或轻量级代码，并调用相关程序来执行，就实现了我们的目的

如何告诉LLM各类HEP程序包的使用方法？

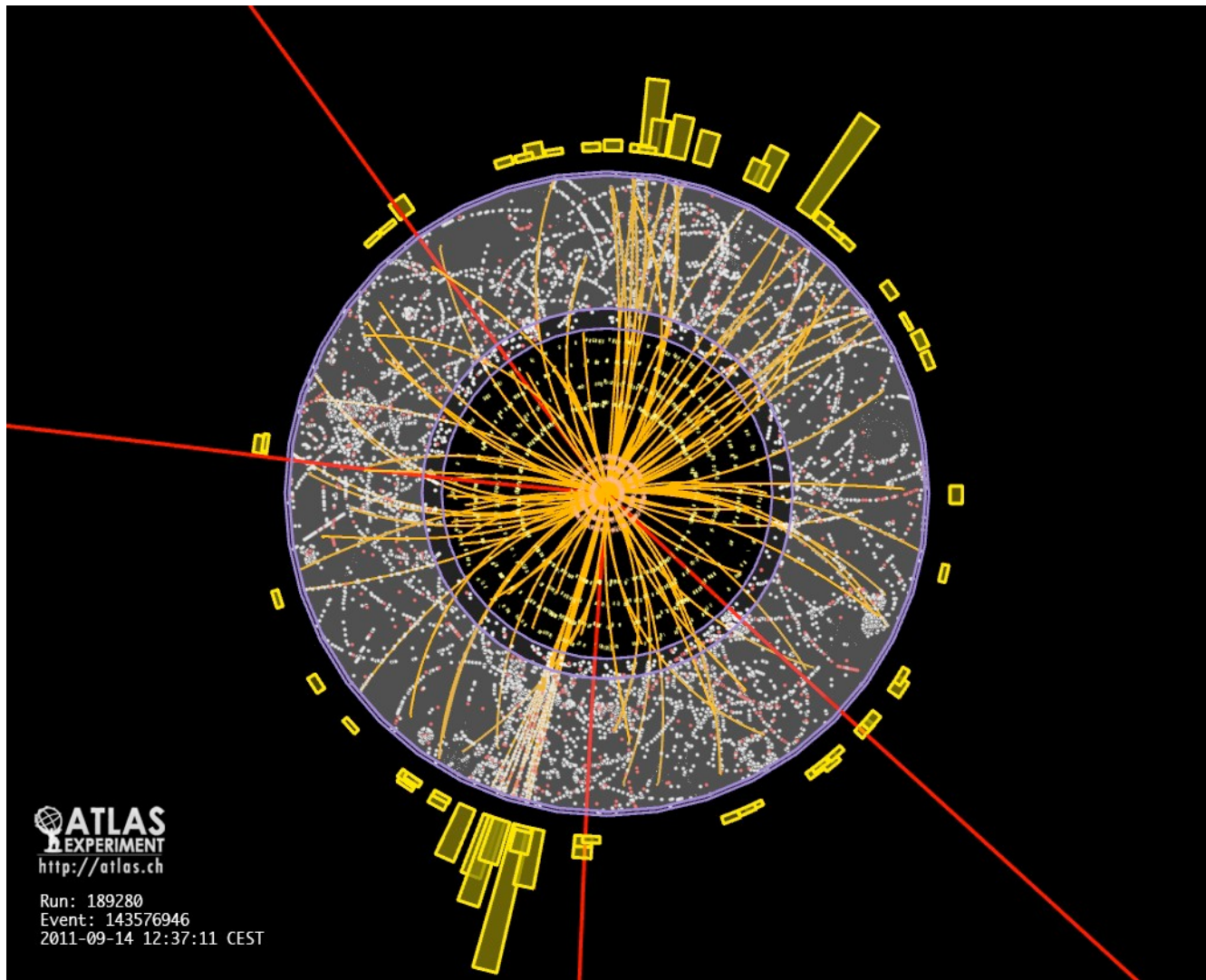
- 1. 当前LLM已经具备了足够的编程能力
- 2. LLM的训练集中已经包含了部分HEP程序的手册和文档
- 3. 通过上下文工程来告诉LLM程序的执行步骤，关键的约定

Agent Skills

# 对撞机物理和相关程序包



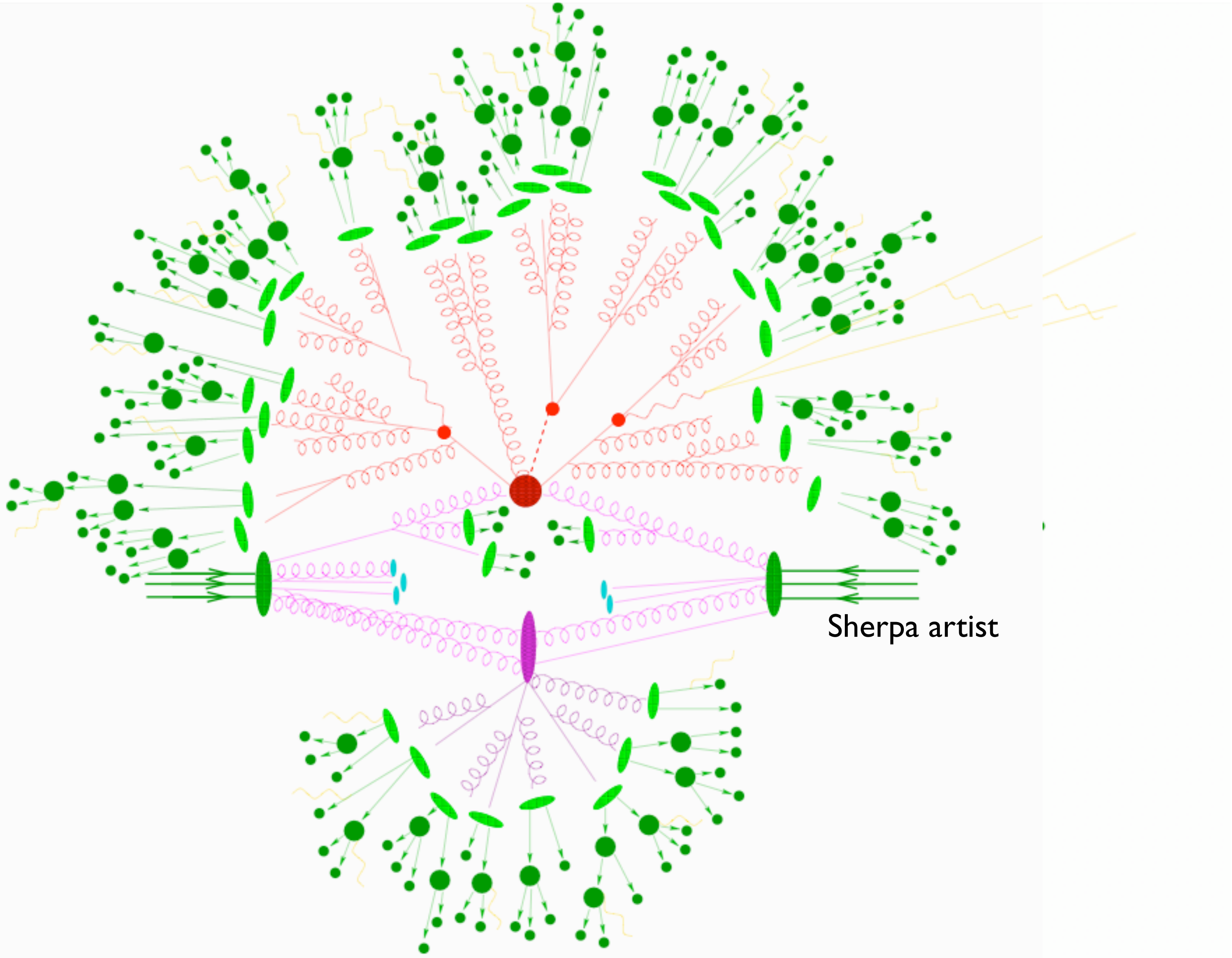
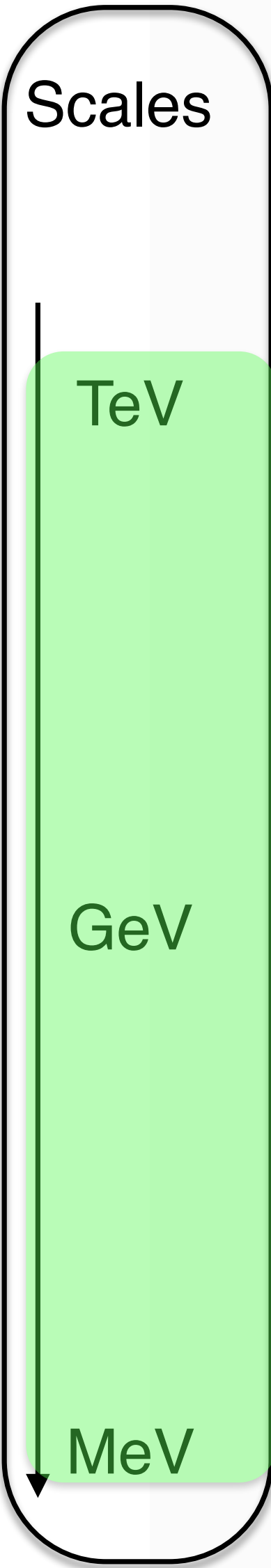
Monte-Carlo Physics



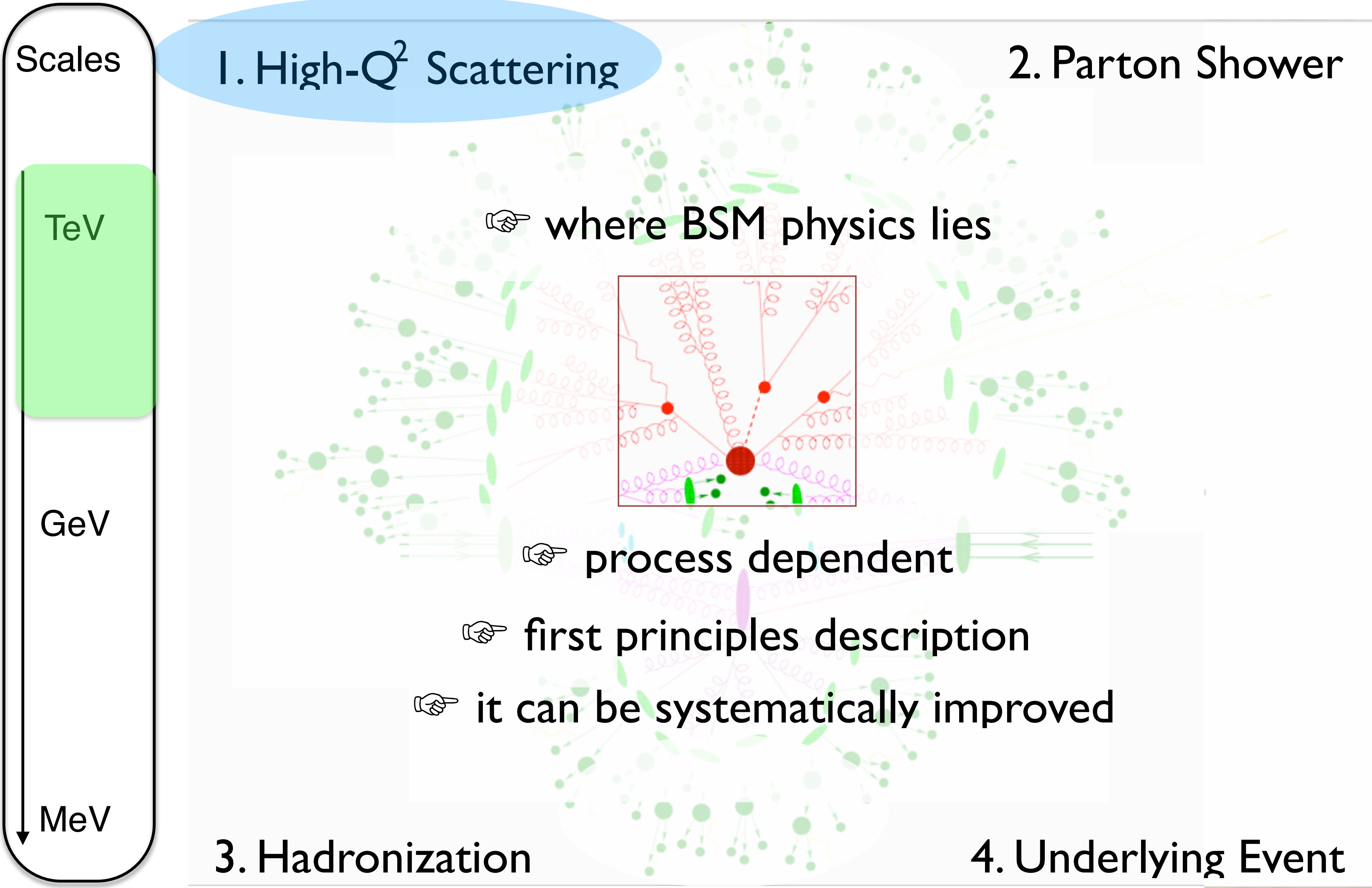
From Olivier Mattelaer's lecture



# 对撞机物理和相关程序包

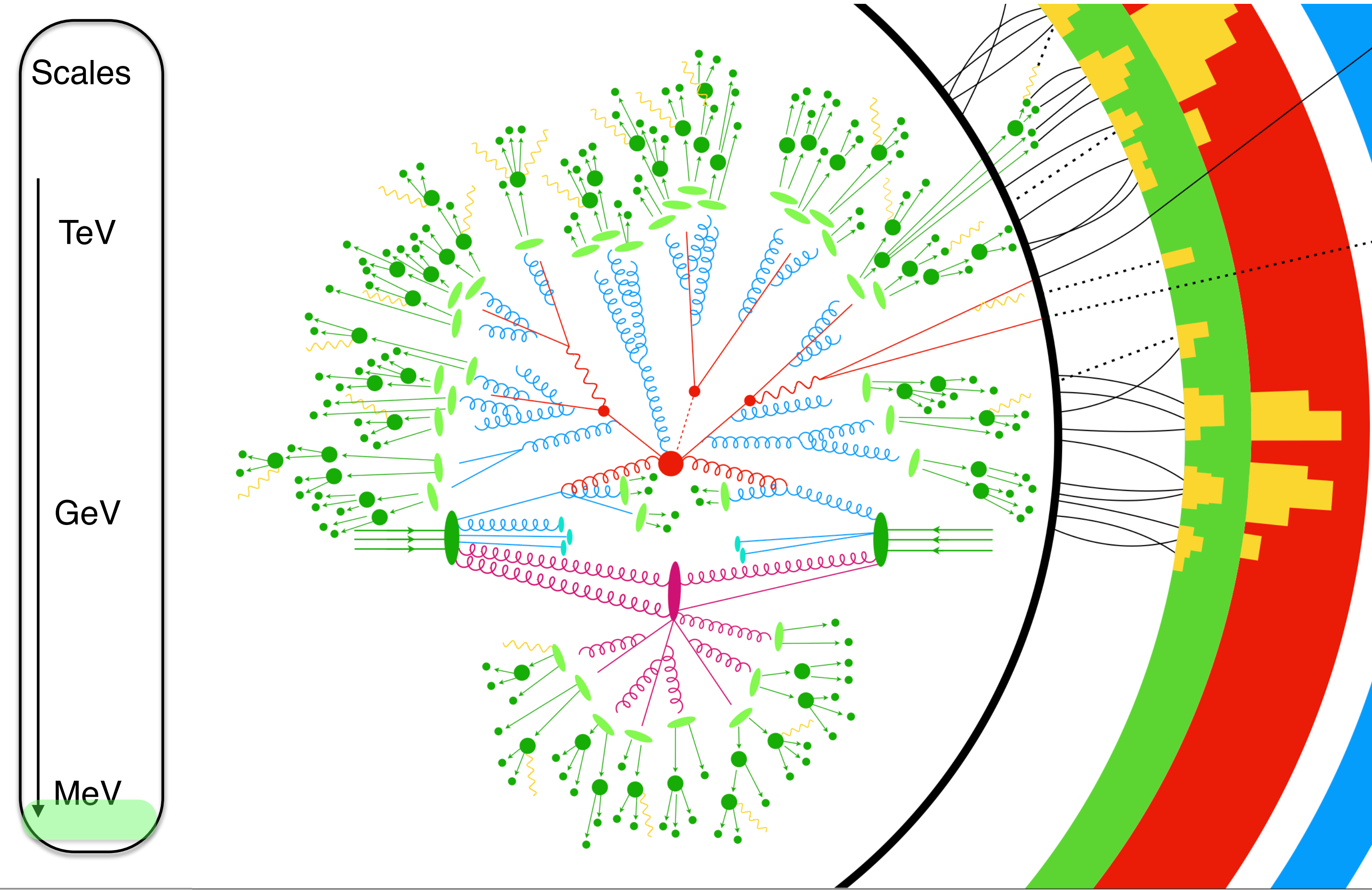


# 对撞机物理和相关程序包

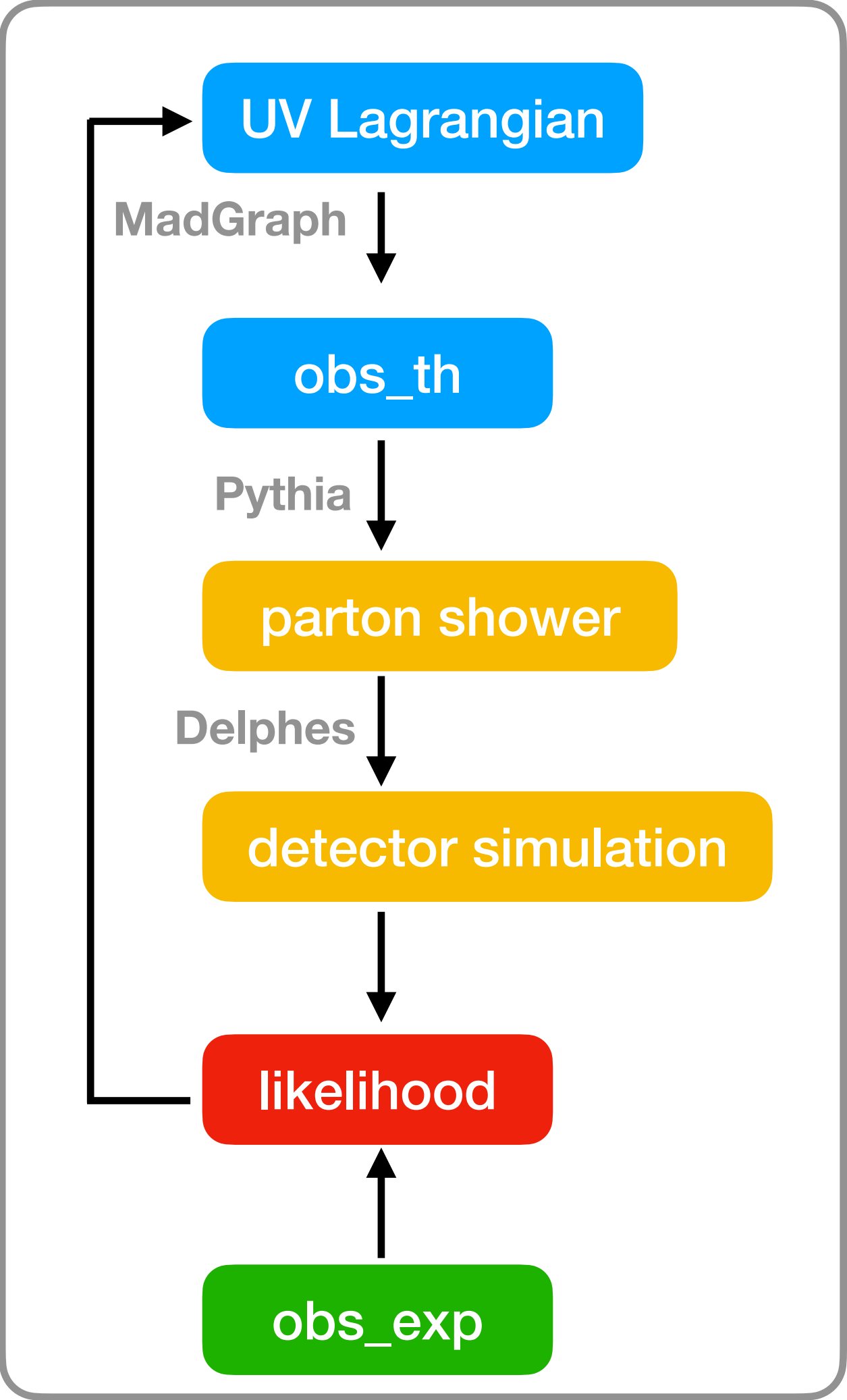




# 对撞机物理和相关程序包

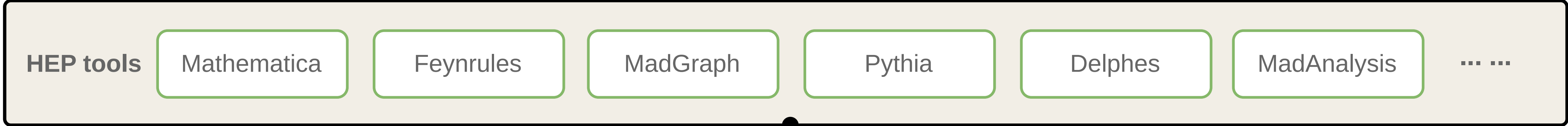


workflow of collider study



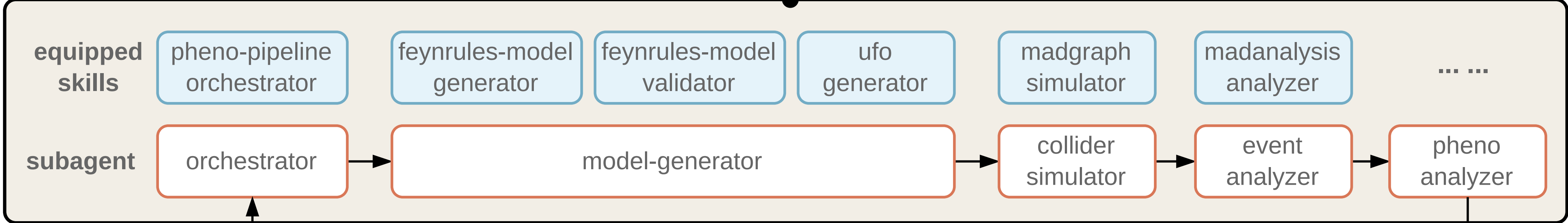
# 适配高能物理的智能体架构

## Execution Backend: Magnus

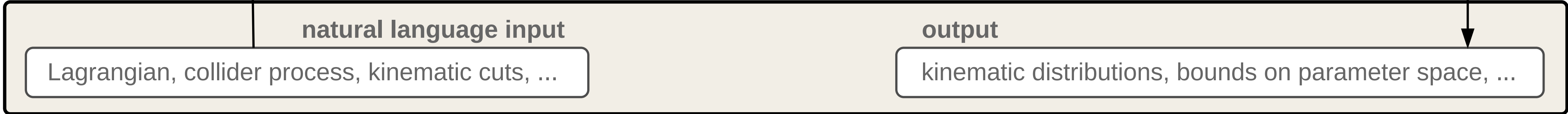


CLI

## Cognitive Reasoning Engine



## User Layer



基于Agent Skills

多智能体

认知-执行解耦



# ColliderAgent

# Hello-World Example

$pp \rightarrow \ell^+ \ell^-$  过程中轻子对的不变质量谱

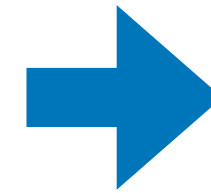


# Hello-World Example

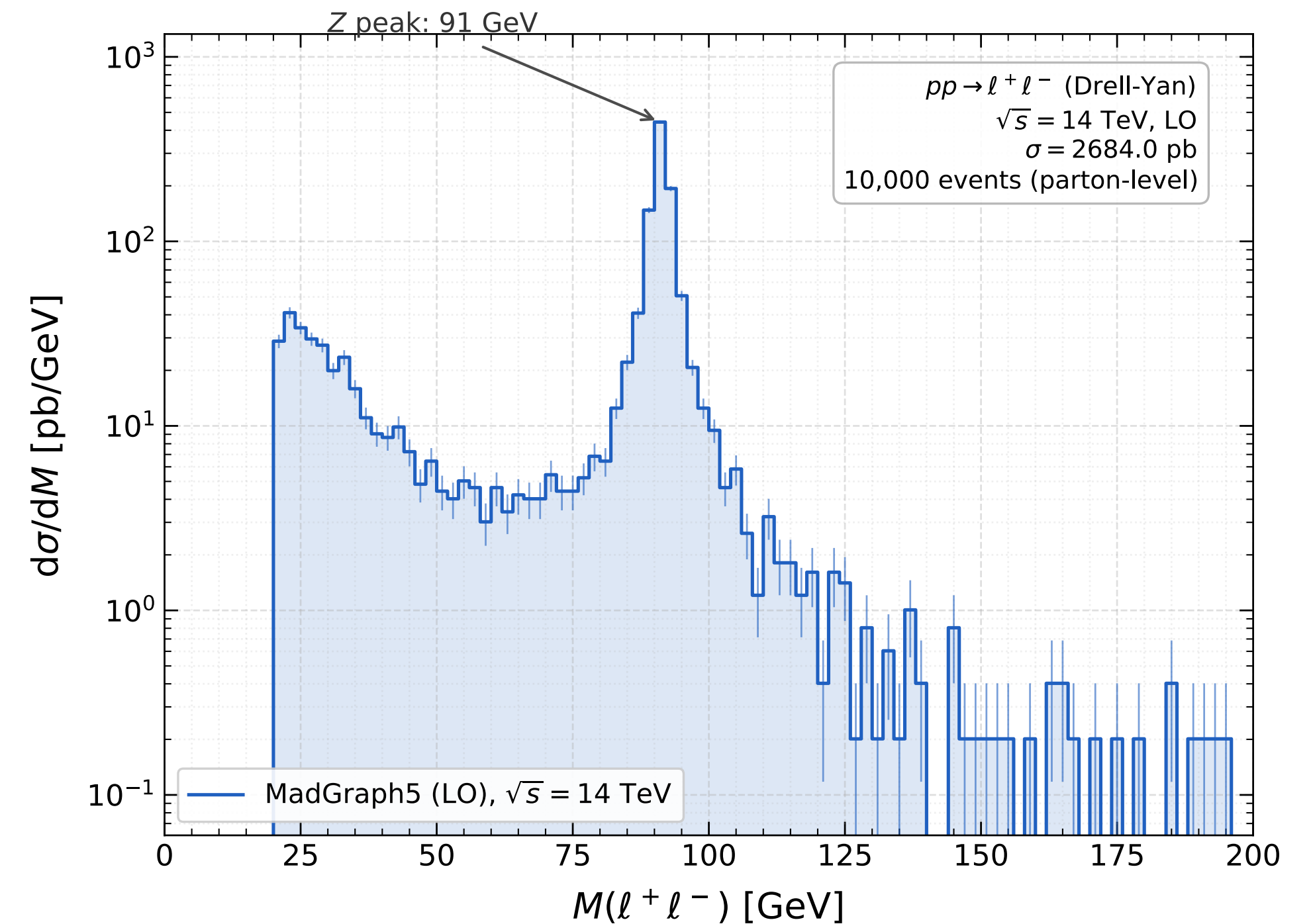
$pp \rightarrow \ell^+ \ell^-$  过程中轻子对的不变质量谱

user prompt

please simulate parton-level  $pp \rightarrow \ell^+ \ell^-$  process at the 14 TeV LHC in the SM, and plot the dilepton invariant mass distribution.

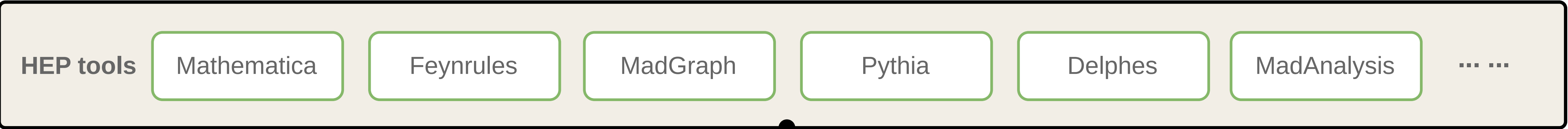


ColliderAgent output

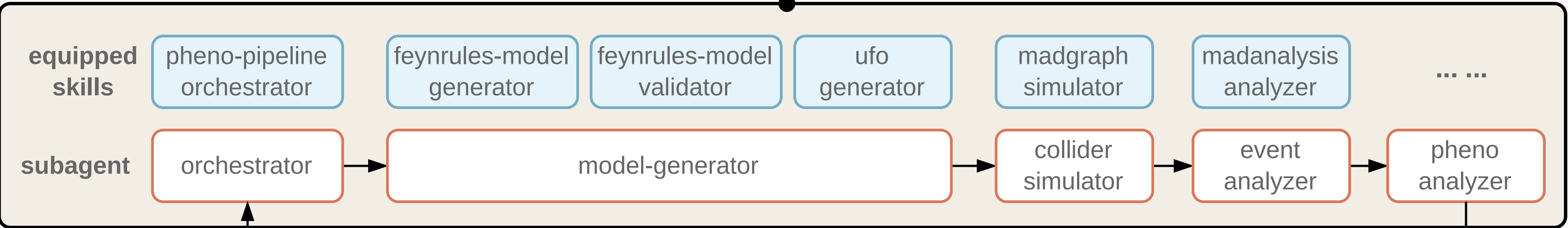


# 智能体架构

## Execution Backend: Magnus



## Cognitive Reasoning Engine



## User Layer



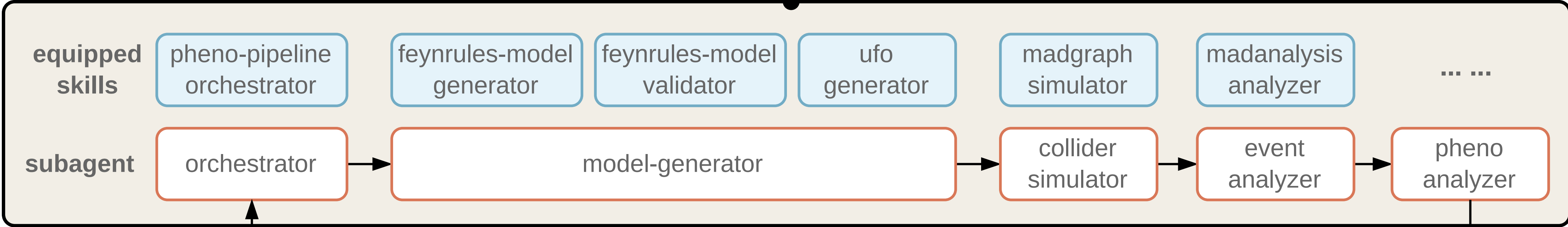
基于Agent Skills

多智能体

认知-执行解耦



# 智能体架构：以Skills为基础



► **Skills**在提示词工程和工具调用的基础上引入了一种结构更完整的知识单元：它是一个可按需动态加载的模块化知识包，内部通常包含操作指令、 workflow 引导、可执行脚本、参考文档与元数据，并以一致、可复现的方式完成对应的专业化操作。这正与对撞机唯象研究的每个阶段相对应，例如从Lagrangian量到.fr模型文件：

```
Workflow

Work through these steps **one at a time**. Do not try to write the complete .fr file in one
shot – copy the skeleton first, then edit each section incrementally.

> ### Step 1: Analyze the Lagrangian ...

> ### Step 2: Map Symbols to FeynRules Conventions ...

> ### Step 3: Create the .fr file from the skeleton ...

> ### Step 4: Define particle classes ...

> ### Step 5: Define parameters ...

> ### Step 6: Write the Lagrangian ...

> ### Step 7: Validate ...
```

```
Step 5: Define parameters

Edit the `M$Parameters` section. For each new coupling constant:
- Set `ParameterType -> External` with `BlockName`, `OrderBlock`, `Value`
- Include `InteractionOrder -> {NP, 1}` for every new physics coupling (both External and Internal)
- Do **NOT** add Mass or Width symbols here – they are already defined by the particle class

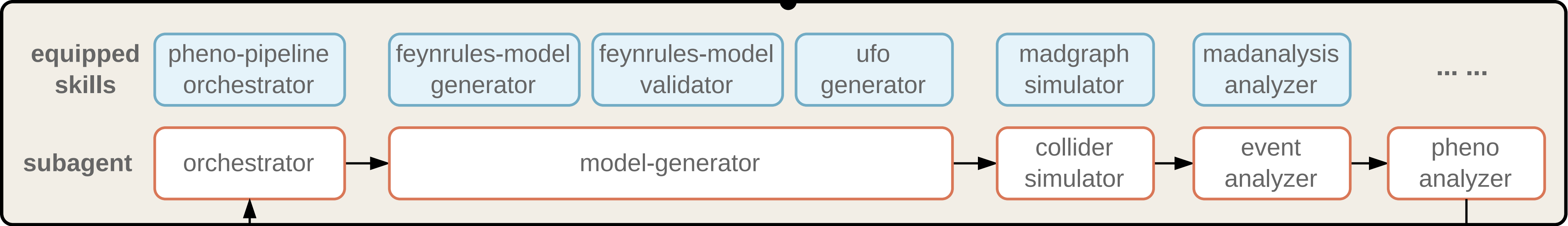
For derived parameters, use `ParameterType -> Internal` with a `Value` expression.

See references/feynrules_syntax.md section 6 for External/Internal parameter
examples, including mixing matrices with `Unitary -> True`.
```

Feynrules-model-generator的部分代码

- **Skills**是通用的行业标准，可以在不同的Agent框架下无缝迁移
- **Skills**可以将唯象研究中的经验（包括一些非正式的、隐性的经验），转化为可存储、可版本管理、可跨项目共享、可持续改进的结构化知识资产。

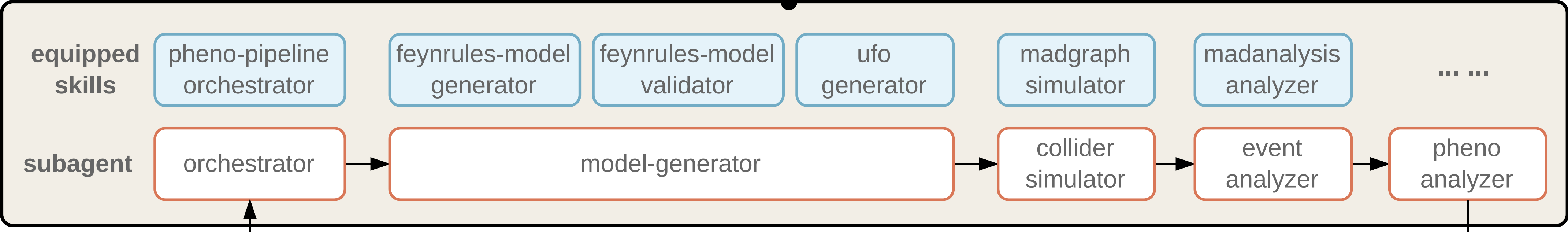
# 智能体架构：以Skills为基础



<code>pheno-pipeline-orchestrator</code>	分解自然语言任务，组织端到端工作流
<code>feynrules-model-generator</code>	把自然语言（包括LaTeX）书写的Lagrangian转换为FeynRules的.fr模型文件
<code>feynrules-validator</code>	检验.fr模型文件：符号约定，FR内置的厄米性等自洽性检验，MadGraph预加载
<code>ufo-generator</code>	从.fr模型文件生成UFO模型文件
<code>madgraph-simulator</code>	编写、执行MadGraph脚本
<code>madanalysis-analyzer</code>	编写、执行MadAnalysis脚本（Normal Mode）
<code>execution-summarizer</code>	生成整个运行流程的总结，以及提示词和代码对应关系的总结。领域专家（或AI）可以便捷的知道ColliderAgent运行的细节，判断执行的对错。
<code>reproduction-guide-generator</code>	生成用于复现运行结果的全部必要脚本（包含.fr文件，.mg5, .ma5等），并生成bash一键运行脚本和复现指南。运行这个脚本后，可以自动化的重复所有的流程，最终复现结果。它保证了ColliderAgent结果的可复现性。
<code>magnus-utility</code>	负责与Magnus平台的作业提交、监控、重跑、结果回收与错误恢复等操作



# 智能体架构： multi-agent

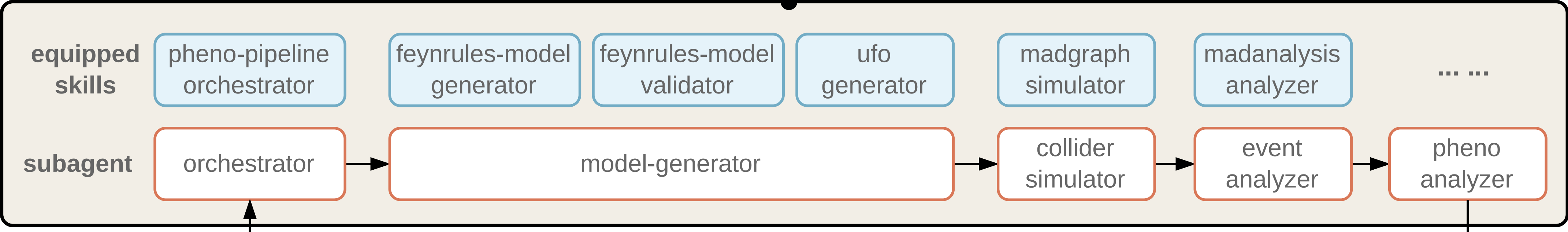


**问题：**对撞机唯象学的流程涵盖模型实现、事件生成、探测器模拟与末态统计分析等多个阶段。每个阶段的关注点、调用的工具以及所需的推理方式均存在显著差异。如果由单一智能体全程承担，在实践中容易导致两类问题：其一是**上下文积累引发的干扰**。模型实现阶段产生的大量技术细节会持续占据智能体的上下文窗口，在进行统计分析时形成无关信息的干扰，降低推理准确性；其二是**角色混淆导致的推理风格不稳定**。同一个智能体在不同阶段需要分别扮演理论建模者、软件工程师与数据分析师的角色，这种频繁切换会使其输出质量趋于不稳定。

**解决：**多智能体

**优点：**子智能体只需维护与自身任务相关的上下文，信息负担显著降低，推理稳定性相应提升。

# 智能体架构： multi-agent



**多智能体 优点：**子智能体只需维护与自身任务相关的上下文，信息负担显著降低，推理稳定性相应提升。

**新的问题：**1) 需要设计编排器来理解用户意图并分解任务。2) 子智能体之间需要共享信息。

**解决方案：**1) 编排器通过pheno-pipeline-orchestrator Skills来分解任务；2) 子智能体通过“双层通信”机制传递信息：subagent完成任务后生成包含此任务全部信息的.md文件，同时生成一份精简版信息并连同.md文件的路径传递给其它subagent。

```

 progress
 dy_ll_14tev
 step1_feynrules.md
 step2_madgraph.md
 step3_madanalysis.md
 step4_postprocessing.md
 ! run_manifest.yaml

```

```

File Paths
- **FeynRules model file**:: `/Users/x/Project/agent/ALP_2_v4/ALP_bosonic.fr`
- **UFO model directory**:: `/Users/x/Project/agent/ALP_2_v4/ALP_bosonic_UFO`

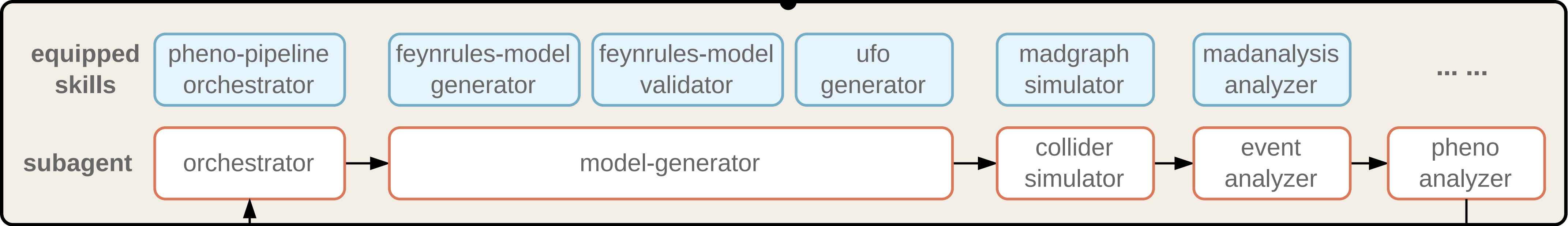
BSM Parameters
| Parameter | Nature | SLHA Block | SLHA Code | Default Value | Description |
|-----|-----|-----|-----|-----|-----|
| `fa` | External | ALPINPUTS | 1 | 1000 (GeV) | ALP decay constant |
| `cWtil` | External | ALPINPUTS | 2 | 1 | Wilson coefficient for SU(2) operator |
| `Malp` | External | MASS | 9000005 | 0.001 (GeV) | ALP mass |
| `gaZA` | Internal | -- | -- | $cWtil * sw / (cw * fa)$ | Effective ALP-Z-gamma coupling |
| `gaZZ` | Internal | -- | -- | $cWtil * (cw^4 - sw^4) / (cw^2 * fa)$ | Effective ALP-ZZ coupling |
| `gaWW` | Internal | -- | -- | $cWtil / fa$ | Effective ALP-WW coupling |

```

step1\_feynrules.md 中的部分内容



# 智能体架构： multi-agent



orchestrator	分解自然语言任务，组织端到端 workflow
model-generator	把自然语言（包括LaTeX）书写的Lagrangian转换为FeynRules的.fr模型文件，并最终生成UFO模型文件，同时校验模型文件的正确性
collider-simulator	负责调用MadGraph生成parton-level事例，调用Pythia进行parton showering，调用Delphes进行探测器模拟
madanalysis-analyzer	调用MadAnalysis（normal mode）来执行事例分析
pheno-analyzer	利用LLM自身能力执行唯象分析任务，包括：profile-likelihood analysis等统计分析，参数空间扫描，画图等。

我们基于Claude Opus 4.6的测试表明：LLM可直接通过Scikit-HEP程序包uproot和awkward来读取LHE、LHCO和ROOT文件，并完成MadAnalysis expert-mode级别的事例筛选任务。

# 智能体架构： 认知-执行解耦

ColliderAgent在系统层面引入了另一项关键设计， **认知-执行分离的双层解耦架构**：

- ▶ 认知层由多智能体系统组成，负责科学任务的理解、分解、规划与结果判断
- ▶ 执行层由统一的计算执行后端Magnus负责，处理作业的实际提交、运行监控与资源管理
- ▶ 两层之间通过清晰定义的CLI接口通信，互不渗透

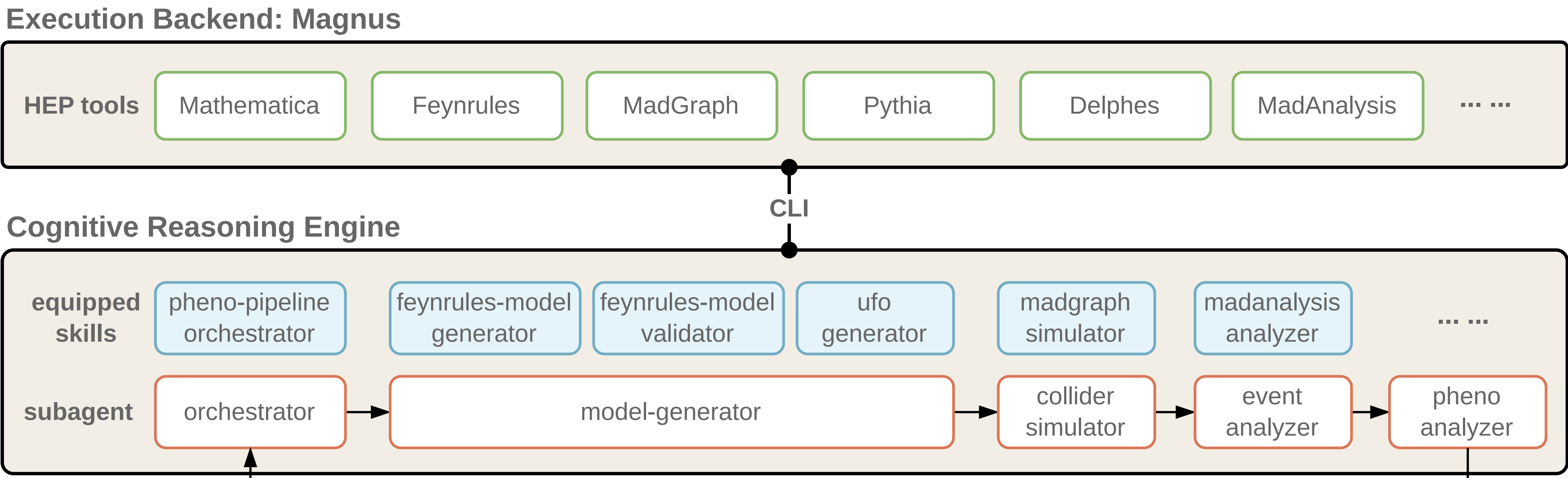
**优点：**

- ▶ 避免了科学逻辑与工程细节的耦合。研究者不再需要将大量精力消耗在配置环境、调试作业等非物理问题上
- ▶ 提升了系统的可移植性与可扩展性。例如，当工具链从MadGraph、Delphes扩展至其他（甚至HEP外的其它领域）软件时，也无需重新设计计算调度机制。

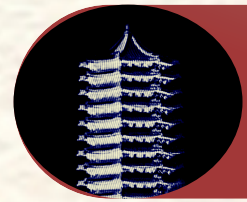
**Magnus：** 为AI时代设计的通用开源计算平台

云原生、持续集成/持续交付的人机协作平台，  
构建高效工作流，沉淀科研蓝图与智能体技能

docker自动化部署  
HPC部署（SLURM）







## Magnus Design Principles

**Magnus** 是一个开源平台，将 HPC 集群转化为人类与 AI 智能体共用的执行后端。围绕执行、沉淀、协作三层架构，遵循三项设计原则：

- **人-智能体对称**：研究者与 AI 智能体通过同一套抽象（Blueprint、Skill、Job）操作平台，所有操作完全可审计，不区分人类点击按钮还是智能体调用 API。
- **蓝图自进化**：Blueprint 不是静态配置文件，而是可被智能体创建、执行、评估和改进的计算原语。一次性实验可以沉淀为 Blueprint，智能体可以基于新结果持续优化它
- **可执行知识图谱**：Blueprint 编码验证过的工作流，Skill 编码领域专家知识，二者构成有向无环的知识图谱，确保可复现性的同时支持持续演进。

协作层（H-H, H-A, A-H, A-A）

\*H = Human, A = Agent

沉淀层（Blueprints, Services, Skills）

执行层（Jobs）



# Magnus 简介

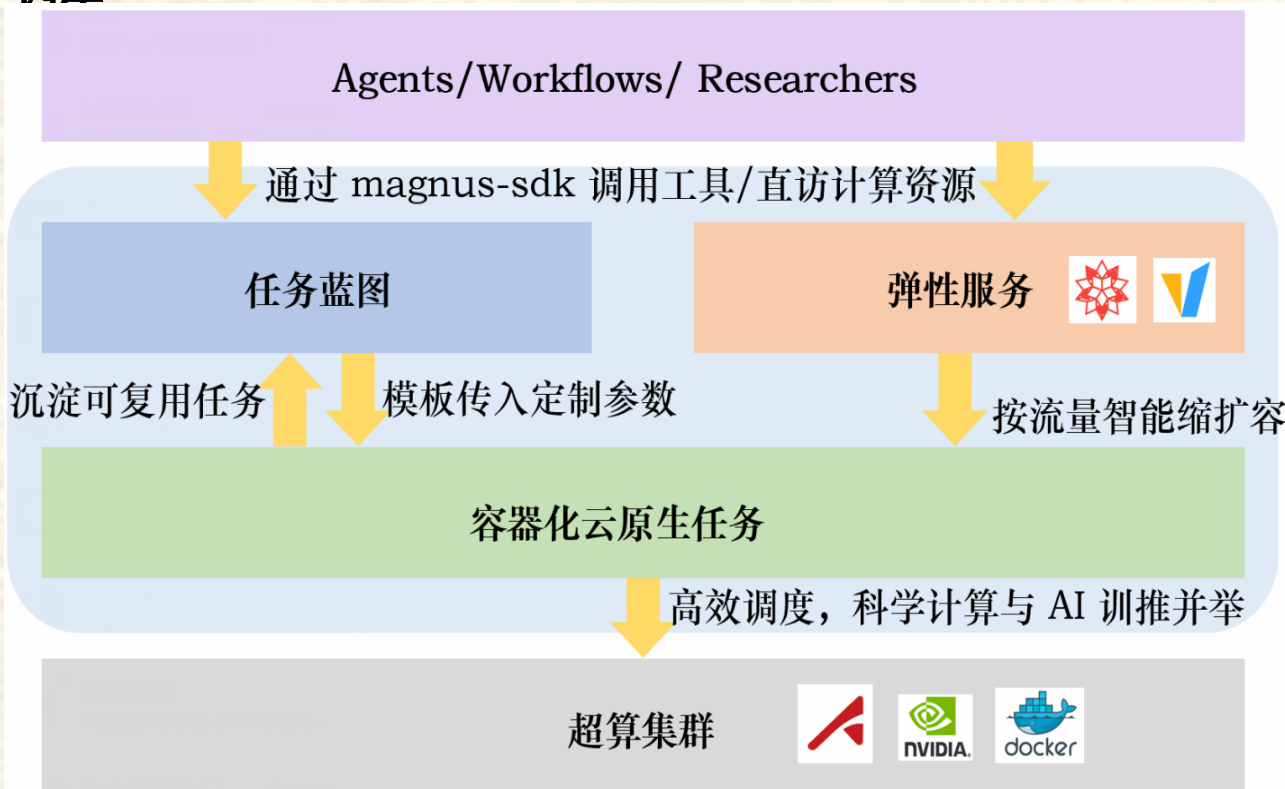


## Magnus 解决方案

集成平台与蓝图。  
集成平台促进人人/人机协作  
蓝图是带类型注解的  
Python 函数；计算流程封装  
为蓝图，即获统一的命令行  
与编程接口，涵盖大型程序  
如 GTC Agent 长程调用等

蓝图 / 蓝图 ID	描述
<b>MadGraph Launch</b> madgraph-launch	基于已编译的 MadGraph5 process directory 运行蒙特卡洛事件生成，产出对撞事件与散射截面数据
<b>MadGraph Compile</b> madgraph-compile	输入 UFO 标准输出和散射过程，枚举费曼图、计算矩阵元，编译生成 MadGraph5 process directory
<b>MadGraph Import Test</b> madgraph-import-test	验证 UFO 模型能否被 MadGraph5 正确 import
<b>Transfer File</b> transfer-file	文件流测试
<b>Generate UFO</b> generate-ufo	从 FeynRules 模型文件生成 UFO 标准输出 自动检测独立模型与 BSM 扩展，支持 .rst 限制文件

四级调度与弹性服务。  
任务分四级优先级，高优  
任务可自动抢占低优任务的  
资源。  
弹性服务空闲时释放资源，  
请求到达时自动唤醒  
超算



技能库。  
技能是多文件知识包，描述领  
域概念、操作规范或最佳实  
践。智能体按需检索并注入上  
下文，团队的领域经验由此沉  
淀为可复用的组织资产。

核心发现（基于Harry论文参数）

1. 波注入期间ECE辐射显著抬升 ~10%
2. 2ωce频率主要来自1-20 keV超热电子（5-10 keV贡献40.3%）
3. 3ωce频率由热电子主导（0-1 keV贡献61.1%）
4. n=0共振线附近分布函数明显变形

plot\_ECE\_weight使用方法

```
bash
基本权重图
python src/plot/plot_ECE_weight.py --data_dir <dir>

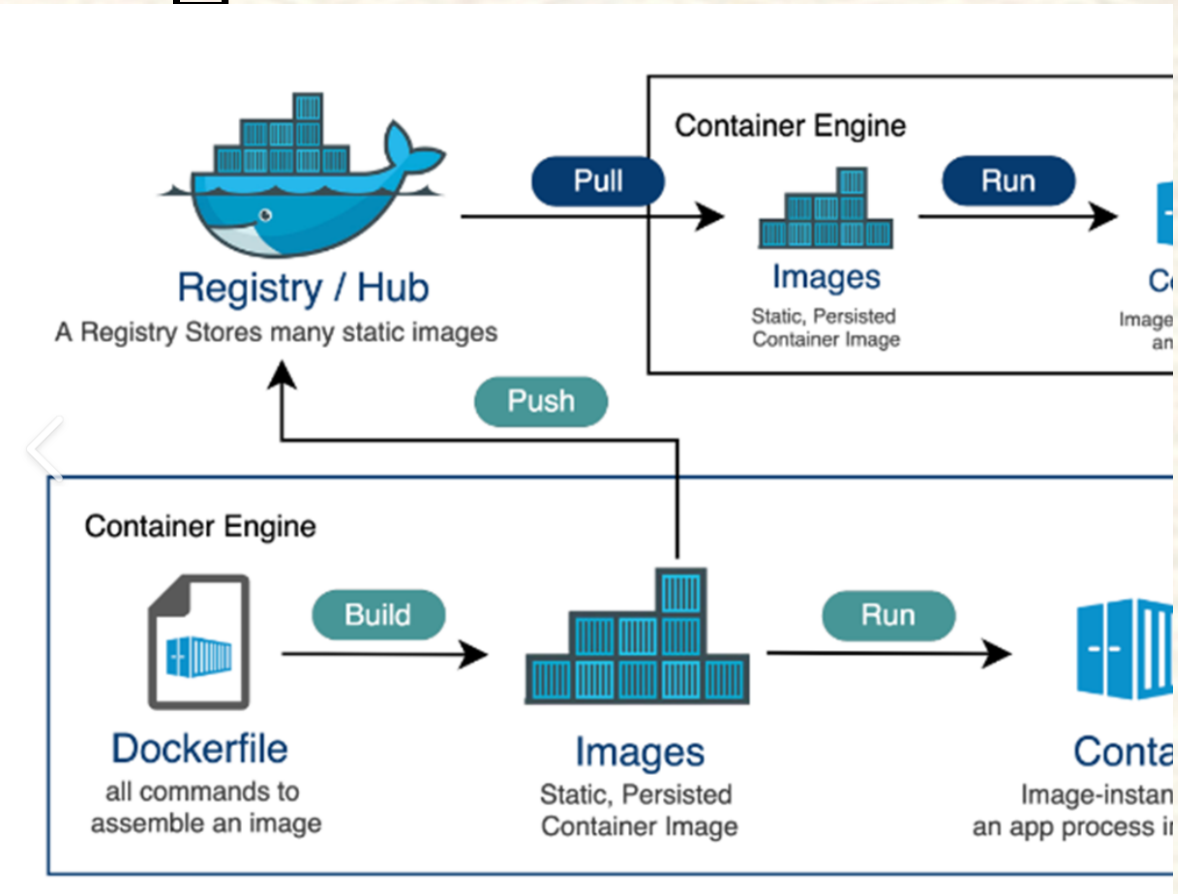
多时间点对比
python src/plot/plot_ECE_weight.py --data_dir <dir> --time_compare 0.9 1.1 1.2 1.4

inject vs nowave ECE差异
python src/plot/plot_ECE_weight.py --data_dir <inject_dir> \
--compare_dir <nowave_dir> --t_compare 1.201 --pmax_plot 2
```

Magnus蓝图使用

```
bash
注入波模拟
```

容器隔离与行为审  
计。所有任务在容器  
内用户态执行，无法  
提权，智能体的每一  
步操作被完整记录，  
人从操作者变为审计  
者



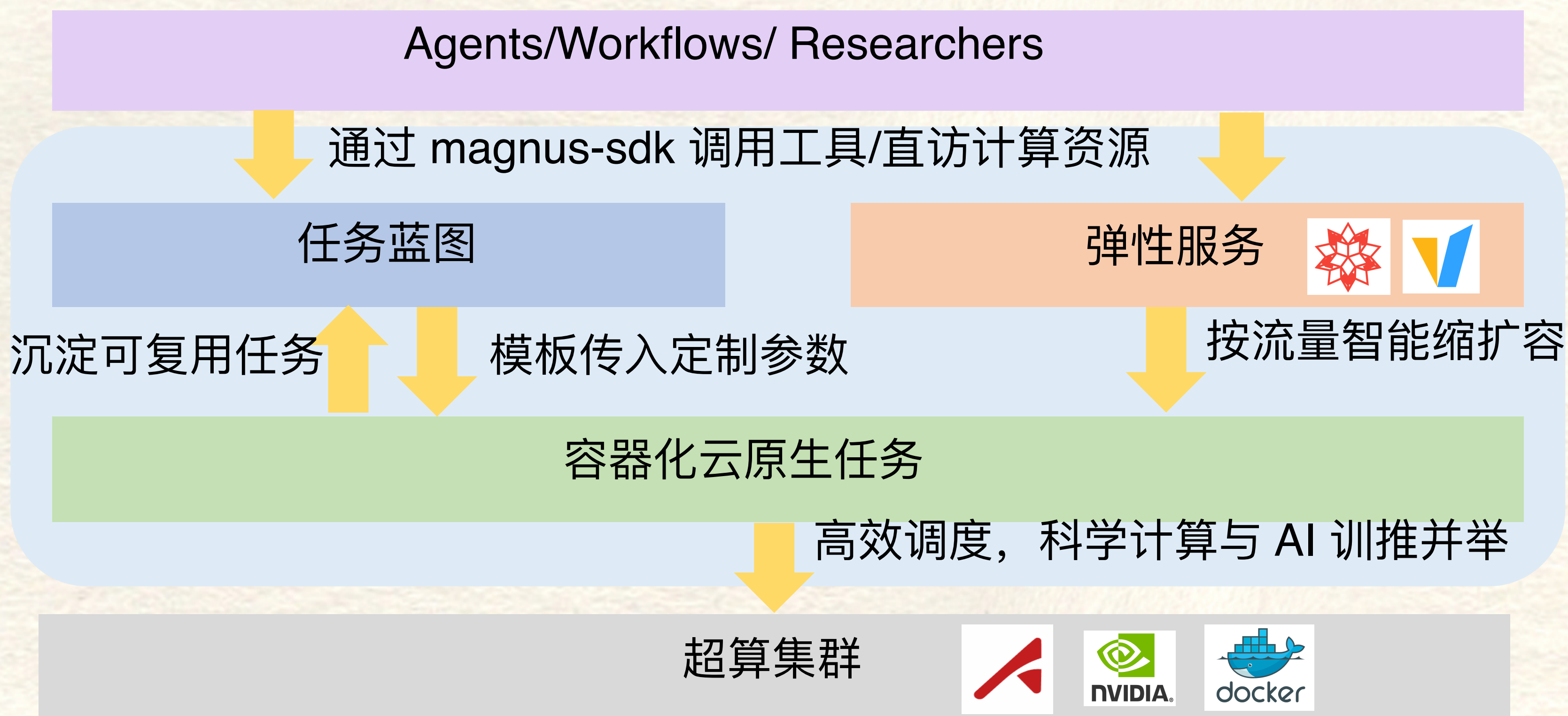


# Magnus 简介

## Magnus: 自动化科学发现的智能体基座

Magnus Platform Next.js 14 TypeScript 5.0+ Python 3.14+ FastAPI 0.104+ PyTorch 2.9+ Scheduler SLURM

云原生、持续集成/持续交付的人机协作平台，构建高效工作流，沉淀科研蓝图与智能体技能



构建人机协作基础

一键复现已有算例，

可进一步沉淀为低代码、参数化的蓝图，

并联研究人员复用与AI智能体调用

原生支持 AI 训推

环境与代码基建开箱即用，

已沉淀 QUADRE 逃逸电子、DESC 仿星

器优化和高能物理对撞模拟程序

多级调度兼顾重保任务与碎片资源

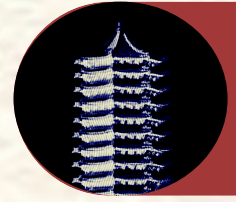
底层兼容性与可行性

无缝对接超算中心和科学计算生态

现已开源：<https://github.com/rise-agi/magnus>；

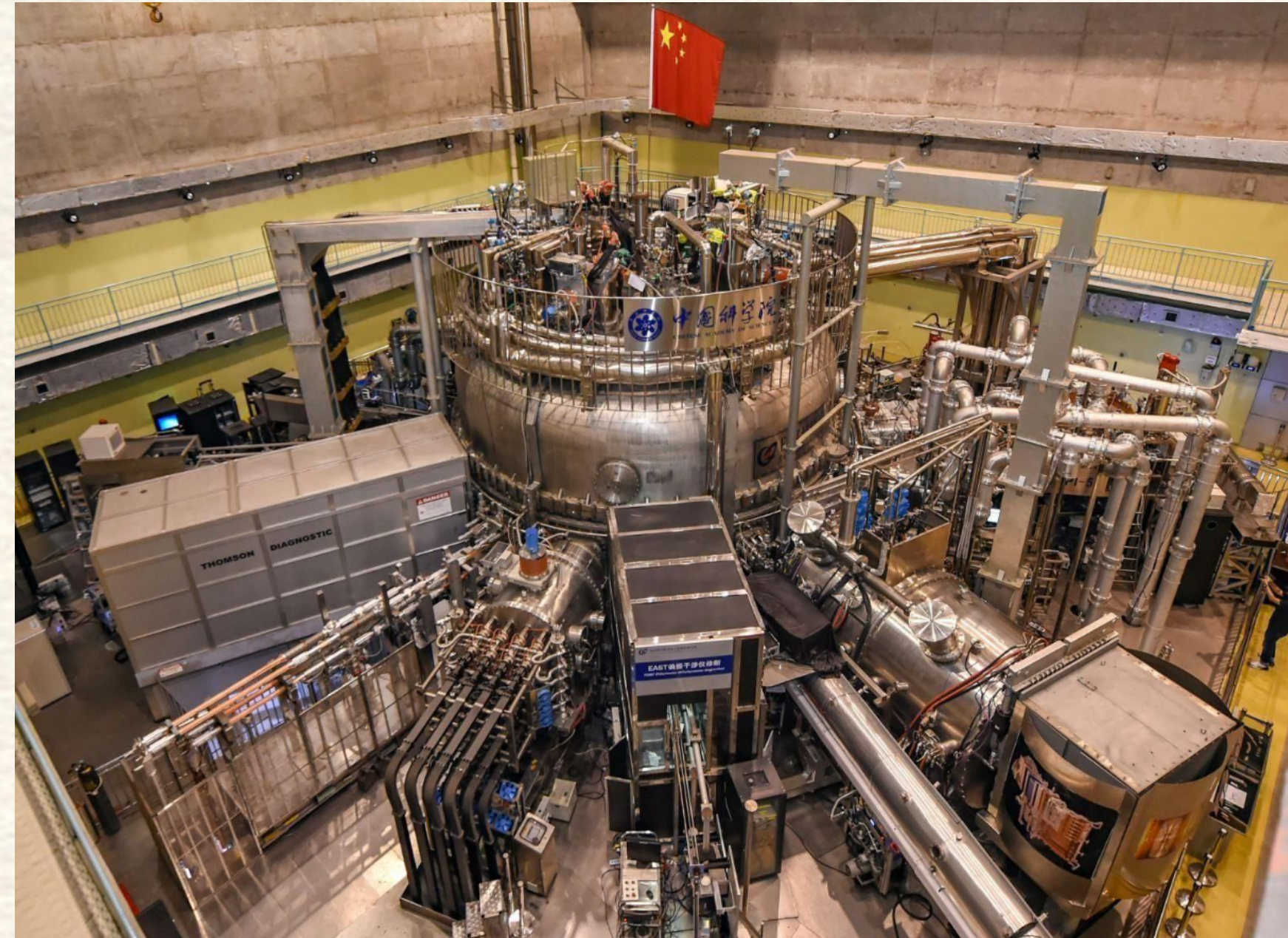


# Magnus 简介：展望



## Magnus, A *To-be* Agentic Infrastructure Automating Scientific Discoveries

**智能体能力的原生集成。** 当前的自然语言入口依赖外部工具链（Claude Code、OpenCode），Magnus 自身尚未具备完整的 agentic loop。下一步是将核心能力——工具调用、上下文工程、多步推理——内化为平台原生能力，降低对单一上游的依赖。



### 从单机多卡到超算集群。

Magnus 目前运行在单节点多 GPU 环境，调度粒度和作业规模受限于此。对接超算将倒逼系统在多节点调度、分布式文件系统和大规模并行作业管理上完成架构升级。

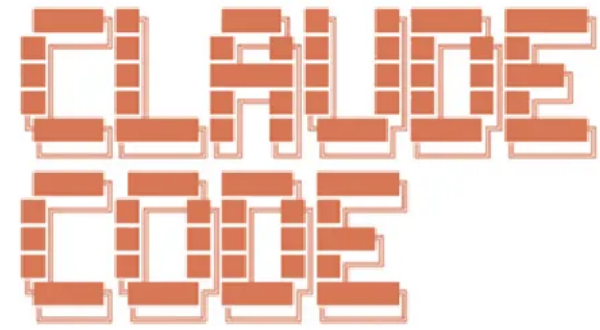
**智能体安全的纵深防御。** 容器隔离限制了执行层的爆炸半径，但提示词注入、上下文投毒等攻击作用于推理层，不在容器的防护范围内。后续需要在系统层面引入输入过滤、权限分级和行为异常检测，形成从推理到执行的多层防线。

**从软件到硬件的语义延伸。** Magnus 的任务目前限于计算任务，但实验物理中的硬件操作——数据采集、设备控制、诊断触发——同样可以复用调度、隔离和审计抽象。将 Magnus job 的语义从“跑一段代码”扩展到“执行一次实验操作”，是平台通向物理实验闭环的关键一步。



# 智能体架构：实现

多智能体+Skills



**Skills:** Cursor, CodeX, OpenCode,  
GitHub Copilot

项目地址： <https://github.com/HET-AGI/ColliderAgent>



# ColliderAgent能做什么？

下面我们通过经典论文的复现来看看ColliderAgent目前具备了哪些能力

# 1. Single LQ production at the LHC

PHYSICAL REVIEW LETTERS **125**, 231804 (2020)

## Lepton-Quark Collisions at the Large Hadron Collider

Luca Buonocore<sup>1,2,\*</sup>, Ulrich Haisch<sup>3,†</sup>, Paolo Nason<sup>4,‡</sup>, Francesco Tramontano<sup>1,§</sup> and Giulia Zanderighi<sup>3,||</sup>

<sup>1</sup>*Dipartimento di Fisica, Università di Napoli Federico II and INFN, Sezione di Napoli, I-80126 Napoli, Italy*

<sup>2</sup>*Physik Institut, Universität Zürich, CH-8057 Zürich, Switzerland*

<sup>3</sup>*Max-Planck-Institut für Physik, Föhringer Ring 6, 80805 München, Germany*

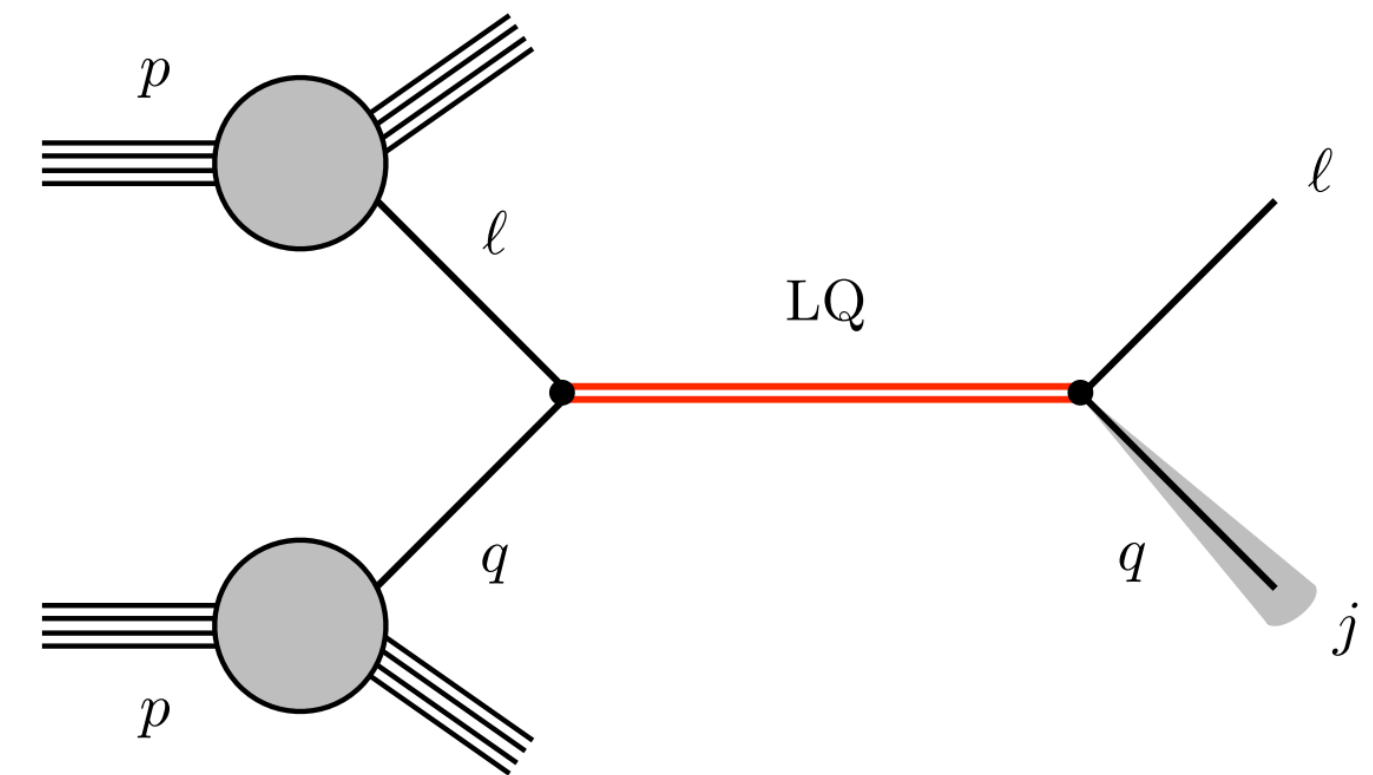
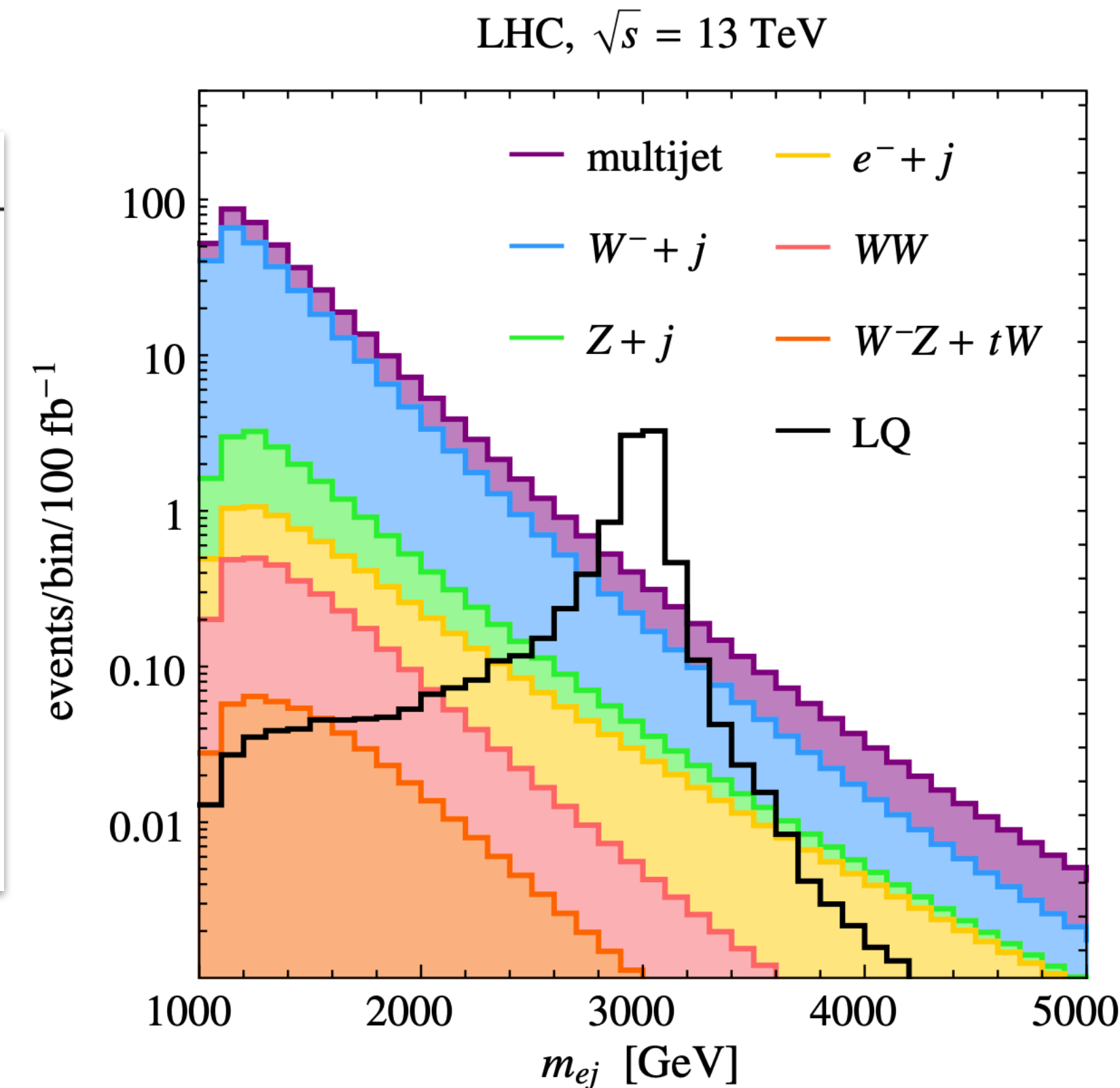
<sup>4</sup>*Università di Milano-Bicocca and INFN, Sezione di Milano-Bicocca, Piazza della Scienza 3, 20126 Milano, Italy*



(Received 29 May 2020; revised 29 September 2020; accepted 9 November 2020; published 2 December 2020)

- ▶ single leptoquark production by picking a lepton from one beam and a quark from the other beam
- ▶ requiring the correct use of the LUXlep PDF set to account for leptons inside the proton.

Since Pythia cannot handle incoming leptonic partons, the initial-state leptons in the LHE events should be replaced by photons before showering. The parton shower backward evolution of PYTHIA then produces only quarks from photon splitting, so after showering the simulation includes initial-state quarks instead of leptons.





# 1. Single LQ production at the LHC

## 1. Target

Considering the scalar leptoquark model described in this document, plot the  $m_{ej}$  distribution for the signal process.

## 2. Scalar Leptoquark Model

### 2.1 Lagrangian

$$\mathcal{L} = \lambda_{eu} \text{LQ}_{eu} \bar{e}_R u_R^c + \text{h.c.}$$

where  $\psi^c \equiv C\bar{\psi}^T$  with  $C = i\gamma^2\gamma^0$  denotes the charge conjugation of  $\psi$  field. Note that the leptoquark  $\text{LQ}_{eu}$  couples to a lepton and a quark (no anti-particles). Here,

- $\text{LQ}_{eu}$  is a scalar leptoquark. It is an  $SU(2)_L$  singlet under the SM gauge group, carrying color triplet and electric charge  $Q = -1/3$ .
- $\lambda_{eu}$  is the Yukawa coupling (real).

### 2.2 Parameters

The benchmark point for the signal:

- mass of LQ:  $M_{\text{LQ}} = 3000 \text{ GeV}$
- total width of LQ:  $\Gamma_{\text{LQ}} = 60 \text{ GeV}$
- $\lambda_{eu} = 1$

user prompt  $\approx$  research note or paper

# 1. Single LQ production at the LHC

## 1. Target

### 3. Collider Simulation

#### 3.1 Process

$pp \rightarrow \text{LQ} \rightarrow e j$ , mediated by the scalar leptoquark, without NWA approximation.

Only consider contributions from the scalar leptoquark. Completely exclude all SM gauge bosons ( $Z, \gamma, W^\pm$ ) and Higgs boson from all Feynman diagrams. Here, the underlying process is a lepton ( $e$ ) from one proton PDF and a quark ( $u$ ) from the other proton fusion to produce the LQ, which then decays back to  $e + j$ . This requires the LUXlep PDF, which provides lepton parton distribution functions inside the proton.

#### 3.2 Collider simulation settings

- Collider: 13 TeV LHC
- Event number: 100000
- Parton shower: Pythia8
- Detector simulation: Delphes with ATLAS card (anti- $k_T$  jets with  $R = 0.4$ )
- PDF: LUXlep; redefine the proton content to include leptons and the photon
- Generation-level cuts:  $p_T(\ell, j) > 500 \text{ GeV}$ ,  $|\eta| < 2.5$
- Store the generated events in LHCO format

#### 3.3 Pythia8 lepton-to-photon workaround

Pythia8 cannot backward-evolve leptons from proton PDFs. The correct steps are:

1. After MadGraph generates the LHE file, replace all initial-state leptons with photons in the LHE file.
2. Disable Pythia8's built-in event validity checks (charge/momentum conservation) so it accepts and showers the manually modified LHE file without rejecting it.
3. Perform shower and detector simulation.

**user prompt  $\approx$  research note or paper**



# 1. Single LQ production at the LHC

## 1. Target

## 3. Collider Simulation

## 4. Numerical Analysis

### 4.1 Event selection

Read the reconstructed events from the Delphes output and apply the following selection cuts:

1. Electron:  $p_T > 500$  GeV,  $|\eta| < 2.5$
2. Jet:  $p_T > 500$  GeV,  $|\eta| < 2.5$  (anti- $k_T$ ,  $R = 0.4$ )
3. Missing transverse energy:  $E_T^{\text{miss}} < 50$  GeV
4. Lepton veto: veto events with additional leptons ( $|\eta| < 2.5$ ,  $p_{T,\ell} > 7$  GeV)
5. Jet veto: veto events with additional subleading jets ( $|\eta| < 2.5$ ,  $p_{T,j} > 30$  GeV)

### 4.2 Signal histogram

Compute the invariant mass  $m_{ej}$  of the leading electron and leading jet for events passing all cuts.

- Bin the  $m_{ej}$  distribution in 100 GeV bins from 0 to 5000 GeV.
- Weight each event by  $w = \sigma \times \mathcal{L} / N_{\text{gen}}$ , where  $\sigma$  is the cross section,  $\mathcal{L} = 100 \text{ fb}^{-1}$ , and  $N_{\text{gen}}$  is the total number of generated events.

## 5. Plot Figure

Plot the signal  $m_{ej}$  distribution with solid black line.

1. After MadGraph generates the LHE file, replace all initial-state leptons with photons in the LHE file.
2. Disable Pythia8's built-in event validity checks (charge/momentum conservation) so it accepts and showers the manually modified LHE file without rejecting it.
3. Perform shower and detector simulation.

user prompt  $\approx$  research note or paper

# 1. Single LQ production at the LHC

## 1. Target

## 3. Collider Simulation

## 4. Numerical Analysis

### 4.1 Event selection

Read the reconstructed events from the Delphes output and apply the following selection cuts:

1. Electron:  $p_T > 500$  GeV,  $|\eta| < 2.5$
2. Jet:  $p_T > 500$  GeV,  $|\eta| < 2.5$  (anti- $k_T$ ,  $R = 0.4$ )
3. Missing transverse energy:  $E_T^{\text{miss}} < 50$  GeV
4. Lepton veto: veto events with additional leptons ( $|\eta| < 2.5$ ,  $p_{T,\ell} > 7$  GeV)
5. Jet veto: veto events with additional subleading jets ( $|\eta| < 2.5$ ,  $p_{T,j} > 30$  GeV)

### 4.2 Signal histogram

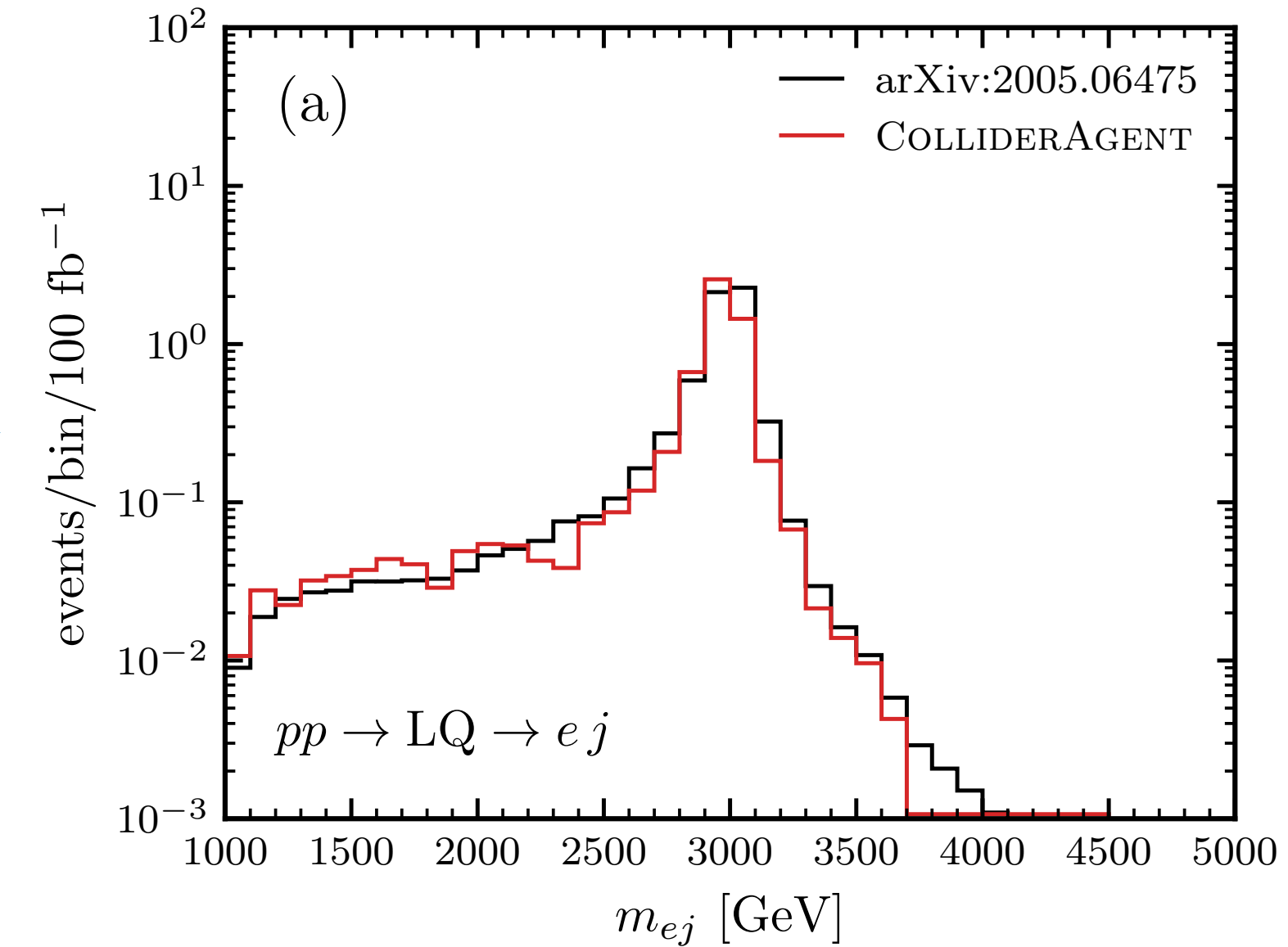
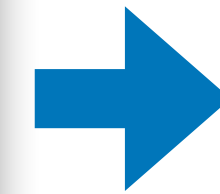
Compute the invariant mass  $m_{ej}$  of the leading electron and leading jet for events passing all cuts.

- Bin the  $m_{ej}$  distribution in 100 GeV bins from 0 to 5000 GeV.
- Weight each event by  $w = \sigma \times \mathcal{L} / N_{\text{gen}}$ , where  $\sigma$  is the cross section,  $\mathcal{L} = 100 \text{ fb}^{-1}$ , and  $N_{\text{gen}}$  is the total number of generated events.

## 5. Plot Figure

Plot the signal  $m_{ej}$  distribution with solid black line.

1. After MadGraph generates the LHE file, replace all initial-state leptons with photons in the LHE file.
2. Disable Pythia8's built-in event validity checks (charge/momentum conservation) so it accepts and showers the manually modified LHE file without rejecting it.
3. Perform shower and detector simulation.



user prompt  $\approx$  research note or paper



# 1. Single LQ production at the LHC

1. Target

3. Collider Simulation

4. Numerical Analysis

4.1 Event selection

Read the reconstructed events from the Delphes output and apply the following selection cuts:

1. Electron:  $p_T > 500$  GeV,  $|\eta| < 2.5$

2. Jet:  $p_T > 500$  GeV,  $|\eta| < 2.5$  (anti- $k_T$ ,  $R = 0.4$ )

3. Missing transverse energy:  $E_T^{\text{miss}} < 50$  GeV

4. Lepton veto: veto events with additional leptons ( $|\eta| < 2.5$ ,  $p_{T,\ell} > 7$  GeV)

5. Jet veto: veto events with additional subleading jets ( $|\eta| < 2.5$ ,  $p_{T,j} > 30$  GeV)

4.2 Signal histogram

Compute the invariant mass  $m_{ej}$  of the leading electron and leading jet for events passing all cuts.

• Bin the  $m_{ej}$  distribution in 100 GeV bins from 0 to 5000 GeV.

• Weight each event by  $w = \sigma \times \mathcal{L} / N_{\text{gen}}$ , where  $\sigma$  is the cross section,  $\mathcal{L} = 100 \text{ fb}^{-1}$ , and  $N_{\text{gen}}$  is the total number of generated events.

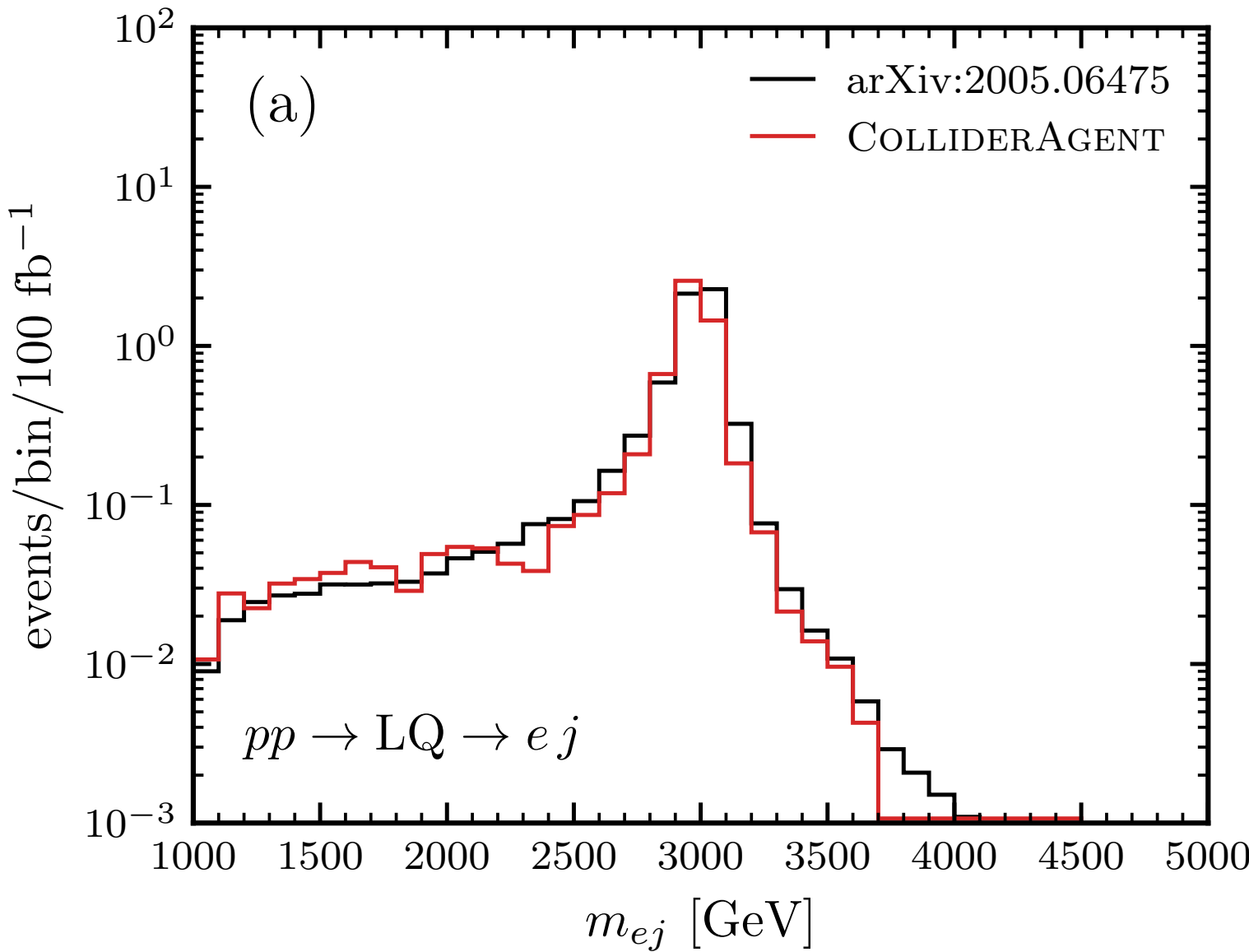
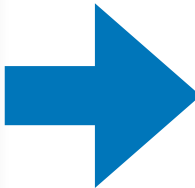
5. Plot Figure

Plot the signal  $m_{ej}$  distribution with solid black line.

1. After MadGraph generates the LHE file, replace all initial-state leptons with photons in the LHE file.

2. Disable Pythia8's built-in event validity checks (charge/momentum conservation) so it accepts and showers the manually modified LHE file without rejecting it.

3. Perform shower and detector simulation.



如果不放心的话，还可以  
“请帮我验证arXiv: 2006.06477中的table 1”

user prompt  $\approx$  research note or paper

SM cross section for lepton-lepton scattering at the LHC

	$e^+\mu^-$	$e^+\tau^-$	$\mu^+\tau^-$	$e^+e^+$	$\mu^+\mu^+$	$\tau^+\tau^+$
$\sigma_{13\text{TeV}}$ [fb]	$0.29^{+0.13}_{-0.10}$	$0.18^{+0.11}_{-0.08}$	$0.16^{+0.10}_{-0.07}$	$0.24^{+0.10}_{-0.08}$	$0.19^{+0.09}_{-0.07}$	$0.08^{+0.06}_{-0.04}$
$\sigma_{27\text{TeV}}$ [fb]	$0.53^{+0.25}_{-0.18}$	$0.34^{+0.21}_{-0.15}$	$0.30^{+0.19}_{-0.14}$	$0.440^{+0.19}_{-0.14}$	$0.34^{+0.16}_{-0.12}$	$0.14^{+0.12}_{-0.07}$

Buonocore, Nason, Tramontano, Zanderighi, 2005.06477

# 2. ALP production at the LHC in EFT

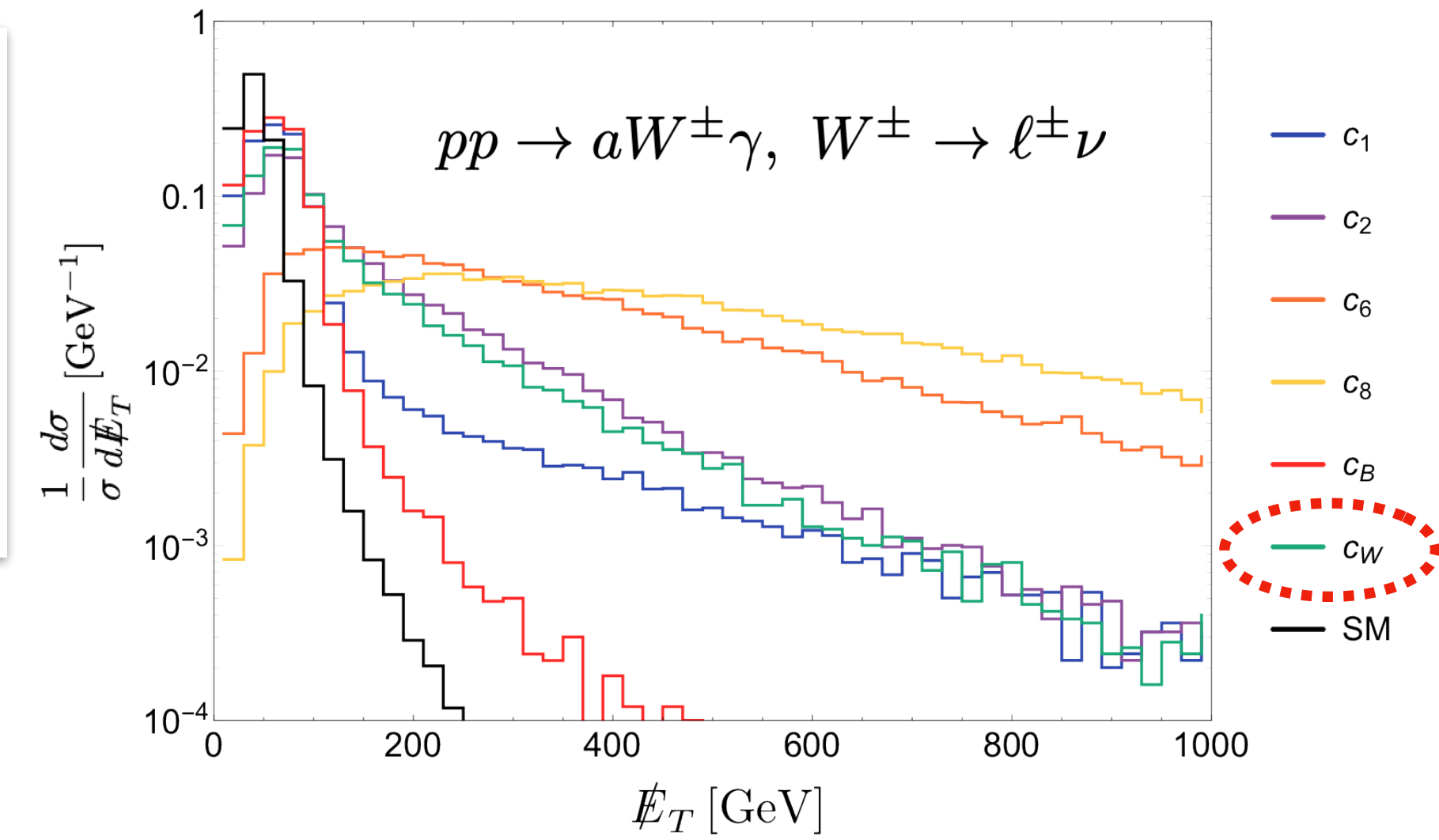
## ALPs Effective Field Theory and Collider Signatures #1

I. Brivio (Bohr Inst. and Madrid, Autonoma U. and Madrid, IFT), M.B. Gavela (Madrid, Autonoma U. and Madrid, IFT), L. Merlo (Madrid, Autonoma U. and Madrid, IFT), K. Mimasu (Louvain U., CP3 and Sussex U.), J.M. No (King's Coll. London and Sussex U.) et al. (Jan 19, 2017)

Published in: *Eur.Phys.J.C* 77 (2017) 8, 572 • e-Print: [1701.05379](https://arxiv.org/abs/1701.05379) [hep-ph]

[pdf](#) [DOI](#) [cite](#) [claim](#)

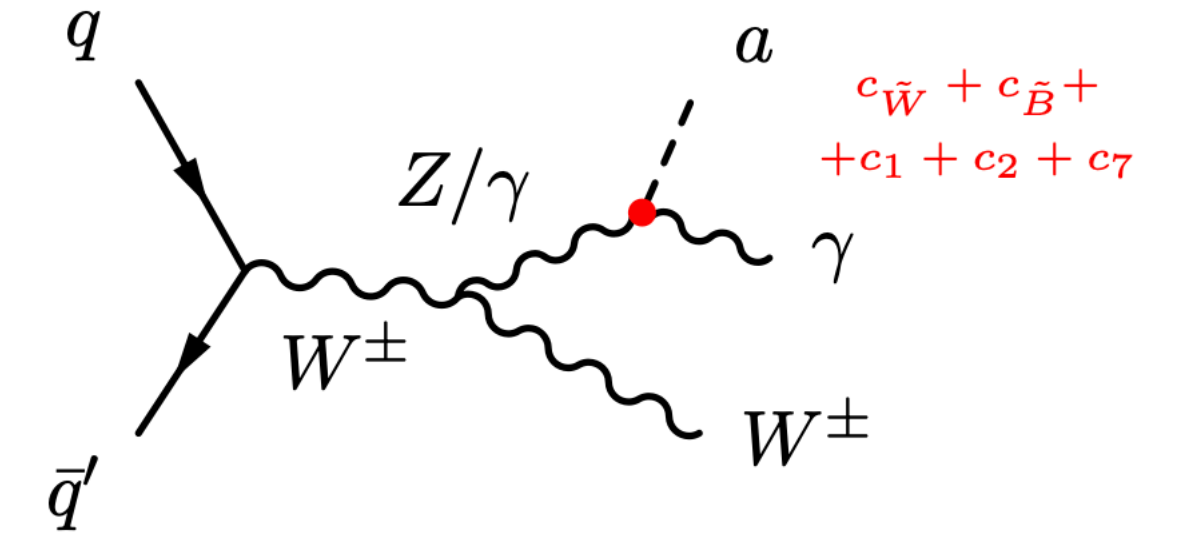
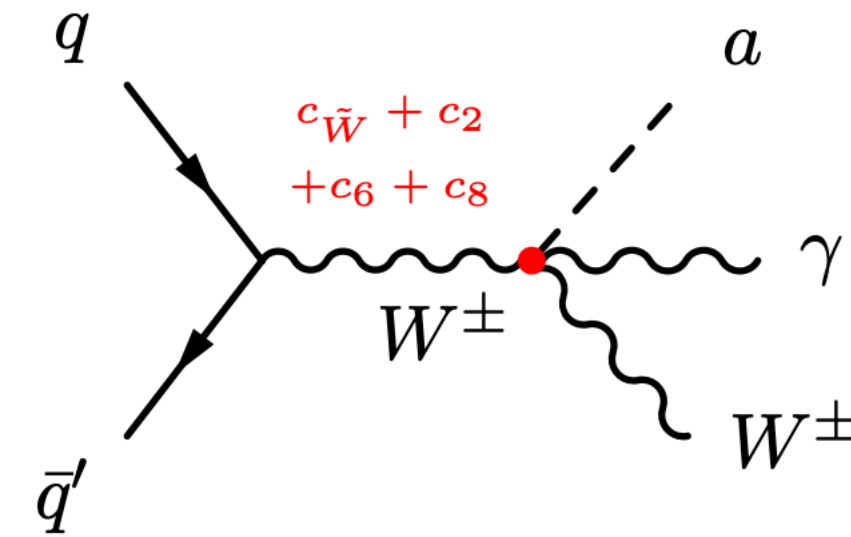
[reference search](#) [324 citations](#)



## EW vertices induced by ALP EFT operators

$$\mathcal{A}_{\tilde{B}} = -B_{\mu\nu} \tilde{B}^{\mu\nu} \frac{a}{f_a} \quad \mathcal{A}_{\tilde{W}} = -W_{\mu\nu}^a \tilde{W}^{a\mu\nu} \frac{a}{f_a}$$

$$c_{\tilde{B}} = -\tan^2 \theta_W c_{\tilde{W}} \text{ so that } g_{a\gamma\gamma} = 0$$





# 2. ALP production at the LHC in EFT

## 1. Target

Considering the ALP EFT in this document, plot the normalized  $E_T^{\text{miss}}$  distribution for  $pp \rightarrow a W^\pm \gamma$  ( $W^\pm \rightarrow \ell^\pm \nu$ ) at  $\sqrt{s} = 13$  TeV LHC.

## 2. ALP EFT

### 2.1 Lagrangian

The ALP bosonic EFT Lagrangian reads:

$$\delta\mathcal{L}_a^{\text{bosonic}} = c_{\tilde{W}} \mathcal{A}_{\tilde{W}} + c_{\tilde{B}} \mathcal{A}_{\tilde{B}}$$

where the operators are

$$\mathcal{A}_{\tilde{B}} = -B_{\mu\nu} \tilde{B}^{\mu\nu} \frac{a}{f_a}, \quad \mathcal{A}_{\tilde{W}} = -W_{\mu\nu}^a \tilde{W}^{a\mu\nu} \frac{a}{f_a}.$$

Here,

- $a$  is the ALP (axion-like particle), a pseudo-scalar singlet, with mass  $m_a$ .
- $f_a$  is the ALP decay constant (dimension of mass).
- $B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu$  is the  $U(1)_Y$  hypercharge field strength tensor.
- $W_{\mu\nu}^a = \partial_\mu W_\nu^a - \partial_\nu W_\mu^a + g\epsilon^{abc}W_\mu^b W_\nu^c$  is the  $SU(2)_L$  weak isospin field strength tensor ( $a = 1, 2, 3$ ).
- $\tilde{B}^{\mu\nu} = \frac{1}{2}\epsilon^{\mu\nu\rho\sigma} B_{\rho\sigma}$ ,  $\tilde{W}^{a\mu\nu} = \frac{1}{2}\epsilon^{\mu\nu\rho\sigma} W_{\rho\sigma}^a$  are the dual field strength tensors.
- $c_{\tilde{W}}, c_{\tilde{B}}$  are dimensionless Wilson coefficients. They satisfy the relation

$$c_{\tilde{B}} = -\tan^2 \theta_W \cdot c_{\tilde{W}}$$

to enforce  $g_{a\gamma\gamma} = 0$ , where  $\theta_W$  is the weak mixing angle.

- In the collider simulation,  $f_a = 1000$  GeV and  $m_a = 0.001$  GeV are chosen. So the free parameter is  $c_{\tilde{W}}$ .

# 2. ALP production at the LHC in EFT

## 1. Target

## 3. Collider Simulation

### 3.1 Process

$$pp \rightarrow a W^\pm \gamma, \quad W^\pm \rightarrow \ell^\pm \nu$$

### 3.2 Collider Simulation Settings

- Collider: 13 TeV LHC
- Event number: 500,000
- Analysis level: Parton-level (no parton shower or detector simulation)
- PDF: nn23lo1
- $f_a = 1000$  GeV,  $m_a = 0.001$  GeV, and  $c_{\tilde{W}} = 1$

---

## 4. Numerical Analysis

### 4.1 Event Selection

Read the LHE events and apply the following selection cuts:

- photon:  $p_T > 20$  GeV,  $\eta < 2.5$
- lepton:  $p_T > 20$  GeV,  $\eta < 2.5$

### 4.2 Histogram and Normalization

- Histogram:  $E_T^{\text{miss}} = |\vec{p}_T^a + \vec{p}_T^\nu|$ , the vector sum of the transverse momenta of all invisible particles (ALP + neutrino).
- Binning: 50 bins, 0–1000 GeV

### 4.3 Plot Figure

Plot the histogram. The height of the histogram is the normalized events, which are defined as the ratio of the number of events in the bin to the total number of events.



# 2. ALP production at the LHC in EFT

## 1. Target

## 3. Collider Simulation

### 3.1 Process

$$pp \rightarrow a W^\pm \gamma, \quad W^\pm \rightarrow \ell^\pm \nu$$

### 3.2 Collider Simulation Settings

- Collider: 13 TeV LHC
- Event number: 500,000
- Analysis level: Parton-level (no parton shower or detector simulation)
- PDF: nn23lo1
- $f_a = 1000$  GeV,  $m_a = 0.001$  GeV, and  $c_{\tilde{W}} = 1$

## 4. Numerical Analysis

### 4.1 Event Selection

Read the LHE events and apply the following selection cuts:

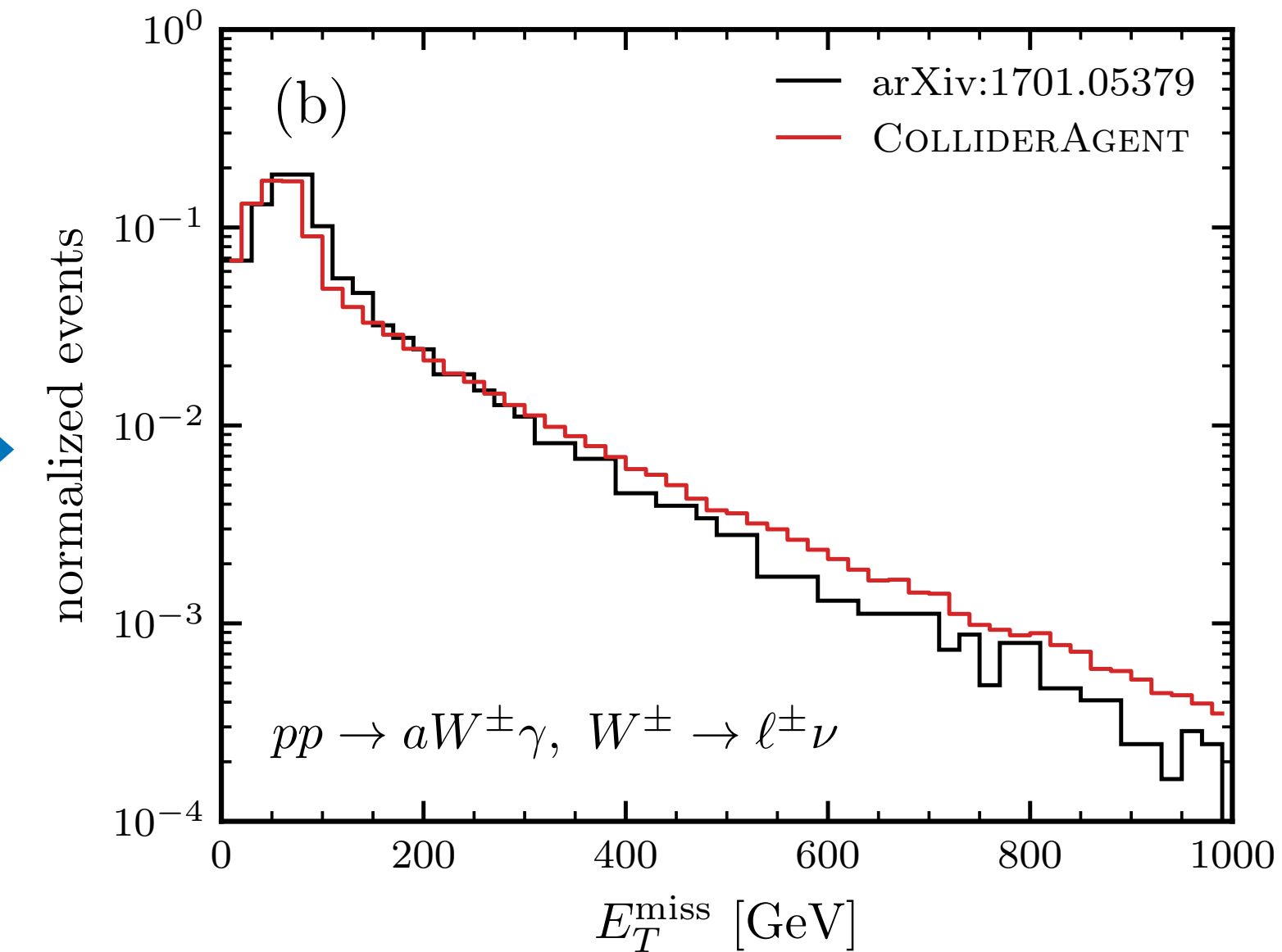
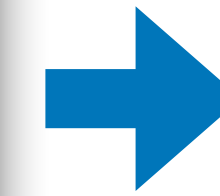
- photon:  $p_T > 20$  GeV,  $\eta < 2.5$
- lepton:  $p_T > 20$  GeV,  $\eta < 2.5$

### 4.2 Histogram and Normalization

- Histogram:  $E_T^{\text{miss}} = |\vec{p}_T^a + \vec{p}_T^\nu|$ , the vector sum of the transverse momenta of all invisible particles (ALP + neutrino).
- Binning: 50 bins, 0–1000 GeV

### 4.3 Plot Figure

Plot the histogram. The height of the histogram is the normalized events, which are defined as the ratio of the number of events in the bin to the total number of events.



# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## Mono- $\tau$ Signatures at the LHC Constrain Explanations of $B$ -decay Anomalies

Admir Greljo (CERN), Jorge Martin Camalich (CERN and IAC, La Laguna and Laguna U., Tenerife), José David Ruiz-Álvarez (Antioquia U.)

Nov 19, 2018

7 pages

Published in: *Phys.Rev.Lett.* 122 (2019) 13, 131803

Published: Apr 6, 2019

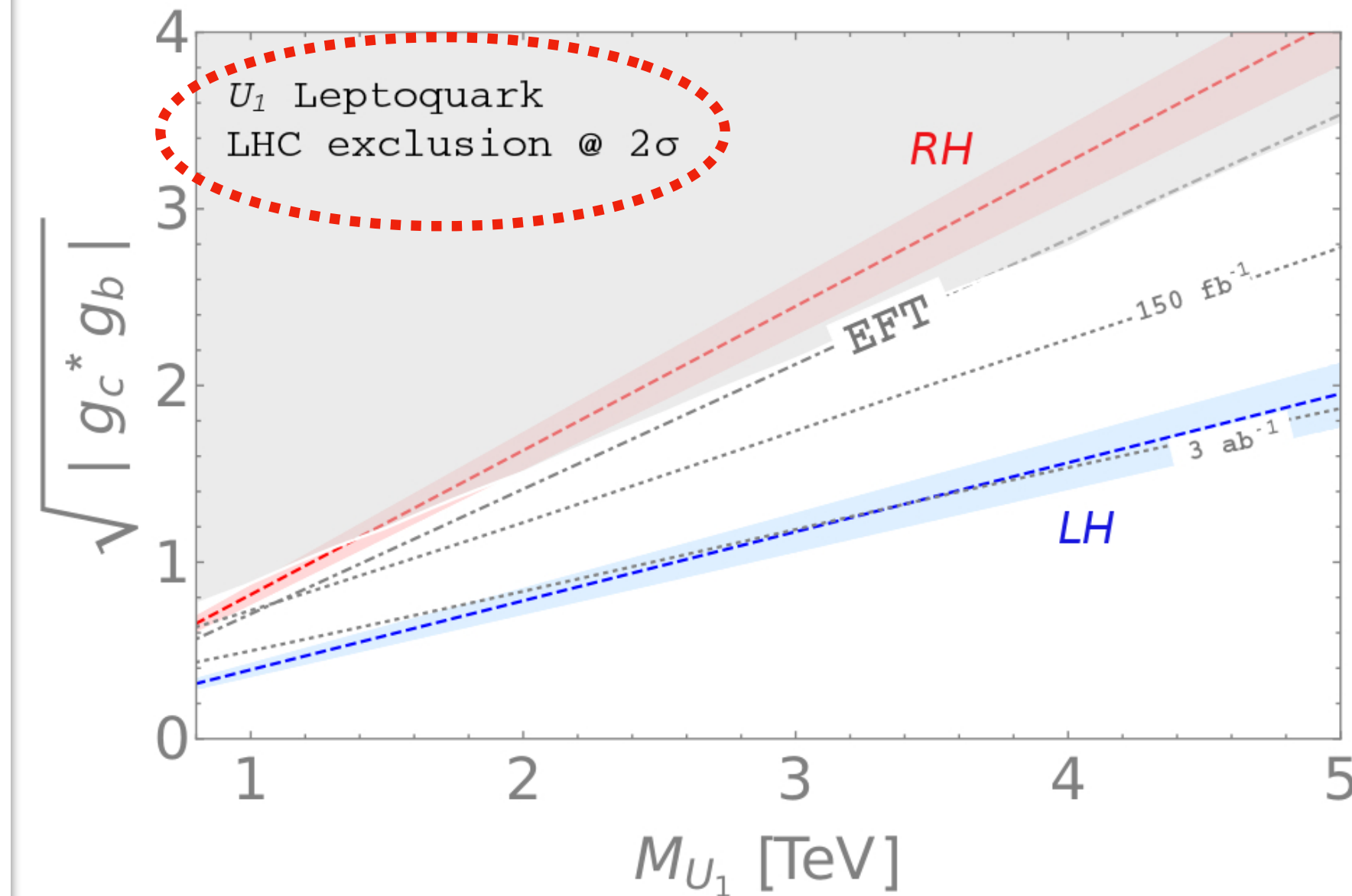
e-Print: [1811.07920](https://arxiv.org/abs/1811.07920) [hep-ph]

DOI: [10.1103/PhysRevLett.122.131803](https://doi.org/10.1103/PhysRevLett.122.131803)

View in: [CERN Document Server](#), [ADS Abstract Service](#)

[pdf](#) [cite](#) [claim](#)

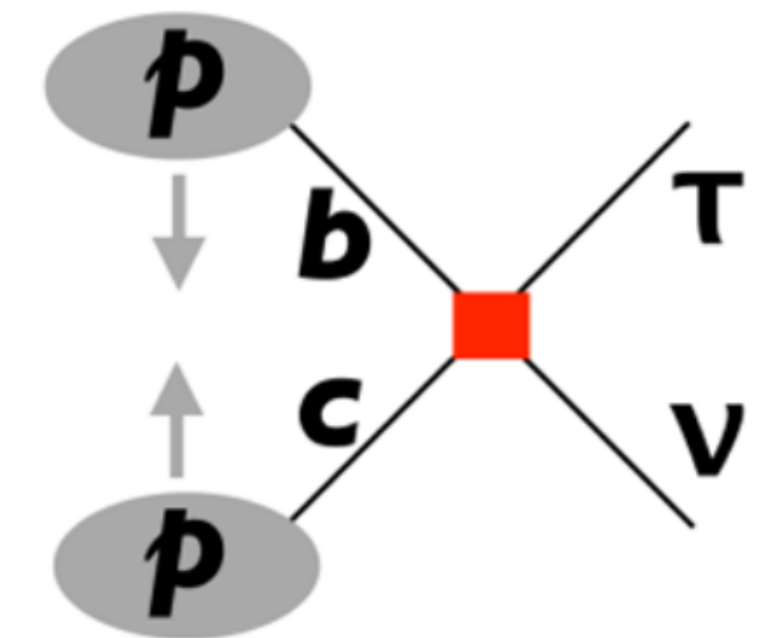
[reference search](#) [135 citations](#)



### ► $U_1$ leptoquark model

$$\mathcal{L}_{U_1} = -(D_\mu U_{1\nu} - D_\nu U_{1\mu})^\dagger (D^\mu U_1^\nu - D^\nu U_1^\mu) + M_{U_1}^2 U_{1\mu}^\dagger U_1^\mu + [g_c(\bar{c}\gamma^\mu P_L \nu_\tau)U_{1\mu}^\dagger + g_b(\bar{b}\gamma^\mu P_L \tau)U_{1\mu}^\dagger + \text{h.c.}]$$

- full workflow, including event generation for the signal process  $pp \rightarrow \tau\nu$ , showering and hadronization, detector simulation with ATLAS and CMS configurations, experiment-specific event selection, and the extraction of the exclusion contour by comparing the resulting signal templates with published LHC data in a profile-likelihood analysis.





# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## 1. Target

Considering the  $U_1$  Leptoquark Model described in this document, plot the  $2\sigma$  exclusion contour for the  $U_1$  leptoquark in the  $(\sqrt{|g_c^* g_b|}, M_{U_1})$  plane, with some other lines and bands.

## 2. $U_1$ Leptoquark Model

In this section, the  $U_1$  Leptoquark Model is introduced.

### 2.1 Lagrangian

$$\mathcal{L}_{U_1} = -(D_\mu U_{1\nu} - D_\nu U_{1\mu})^\dagger (D^\mu U_1^\nu - D^\nu U_1^\mu) + M_{U_1}^2 U_{1\mu}^\dagger U_1^\mu + [g_c (\bar{c} \gamma^\mu P_L \nu_\tau) U_{1\mu}^\dagger + g_b (\bar{b} \gamma^\mu P_L \tau) U_{1\mu}^\dagger + \text{h.c.}]$$

Here,

- $U_1$  is a vector leptoquark beyond the SM. It is a new gauge boson carrying color triplet,  $SU(2)_L$  singlet, and electric charge  $Q = 2/3$ .
- $D_\mu$  is the covariant derivative of the SM gauge fields.

### 2.2 Parameters

The free parameters are:

- $M_{U_1}$  is the mass of  $U_1$ .
- $g_c$  and  $g_b$  are the couplings of  $U_1$  to  $c$  and  $b$  quarks, respectively. Both of them are real.

# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## 1. Target

### 3. Collider Simulation

#### 3.1 Process

$$pp \rightarrow \tau \nu$$

which is mediated by the  $U_1$  vector leptoquark. Since the  $U_1$  leptoquark couples to  $b$  and  $c$  quarks, the initial-state  $b$  and  $c$  partons from the proton PDF must be included.

#### 3.2 Collider simulation settings

We have two runs. For each run,

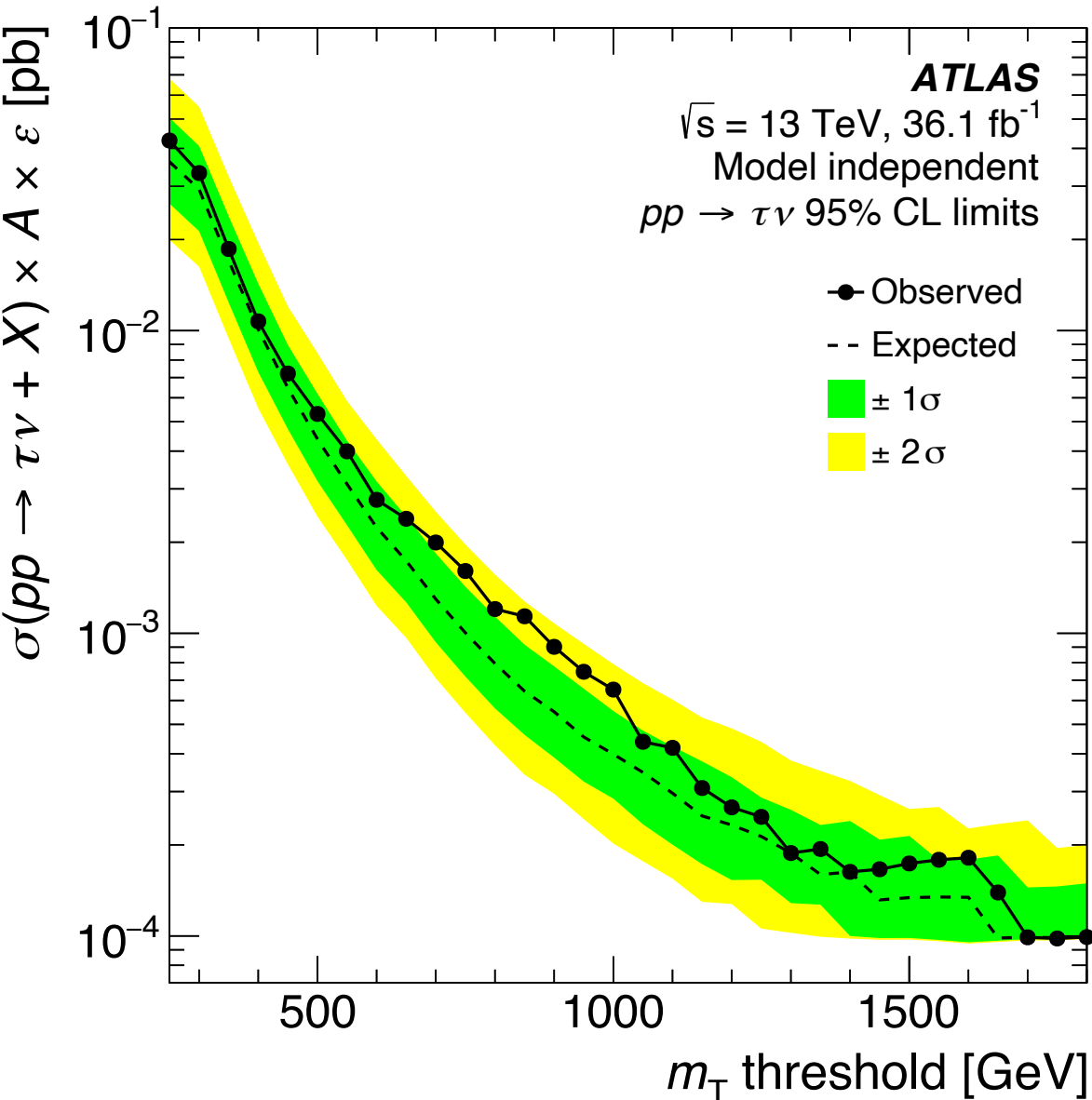
- collider: 13 TeV LHC
- event number: 10000
- parton shower: Pythia8
- perform mass scan for  $M_{U_1}$  masses: 750, 1000, 1250, 1500, 2000, 2500, 3000, 4000, 5000 GeV
- output format for reconstructed events: LHCO
- comment: decay width of LQ is not relevant, since only the  $t$ -channel contribution exists.

Run 1 (with ATLAS detector simulation):

- detector simulation: Delphes ATLAS card

Run 2 (with CMS detector simulation):

- detector simulation: Delphes CMS card





# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## 1. Target

### 3. Collider Simulation

#### 4 Numerical Analysis

##### 4.1 Experimental data

Binned  $m_T$  distribution from ATLAS and CMS are given below.

##### ATLAS

- Stored as `analysis/hepdata/table1.yaml`
- 22 bins from 250 to 3200 GeV (log-spaced)
- Columns: observed events  $n_i$ , SM background  $b_i$ , symmetric error  $\delta b_i$

##### CMS

The experimental data is given below:

$m_T$ bin (GeV)	$n_{\text{obs}}$	$b_{\text{SM}}$	$\delta b$
320–500	1203	1243	160
500–1000	452	485	77
1000–3200	15	23.4	6.2

##### 4.2 Simulated signal events

In this section, the expected signal events in each bin are calculated.

##### step 1: event selection

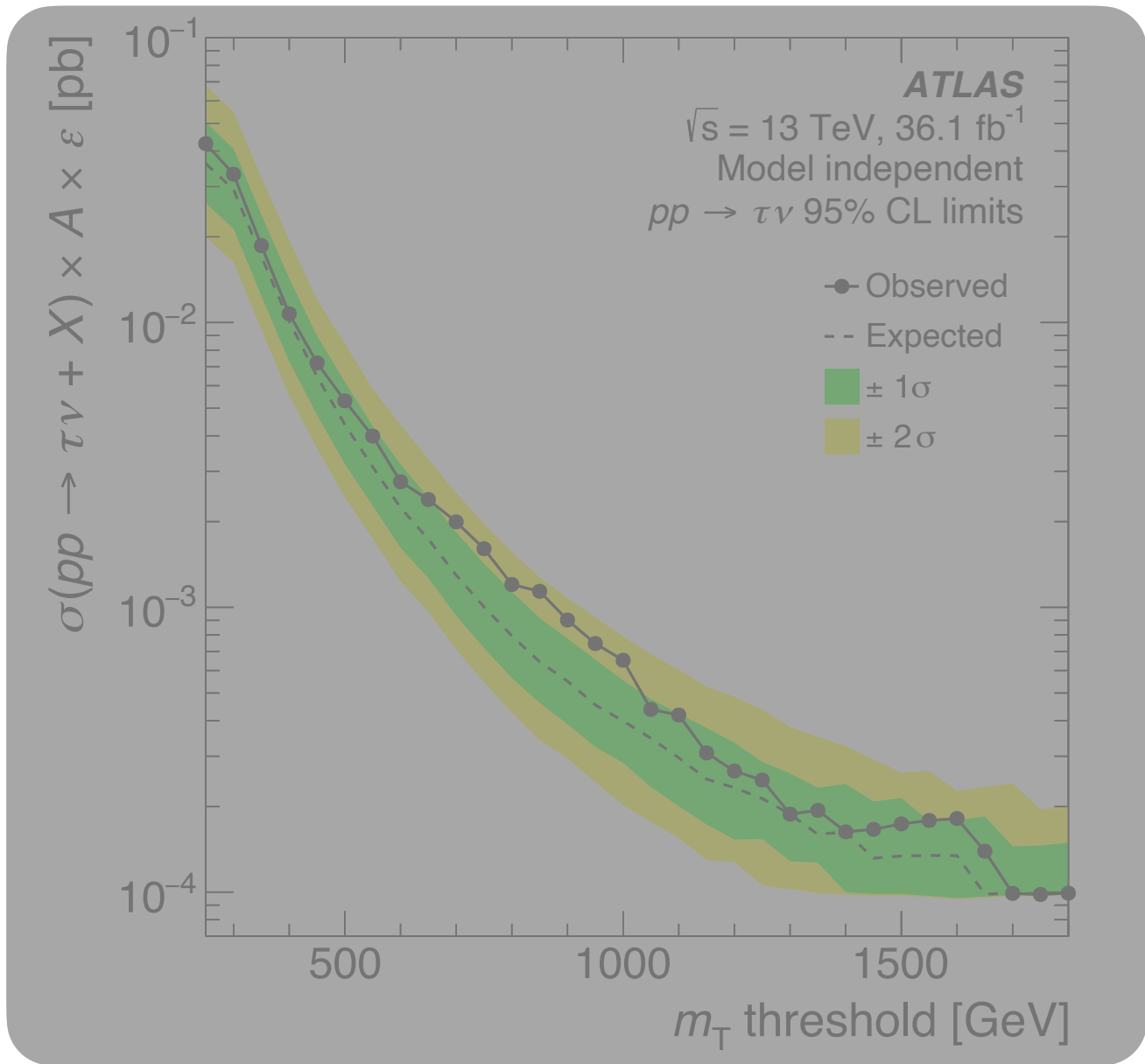
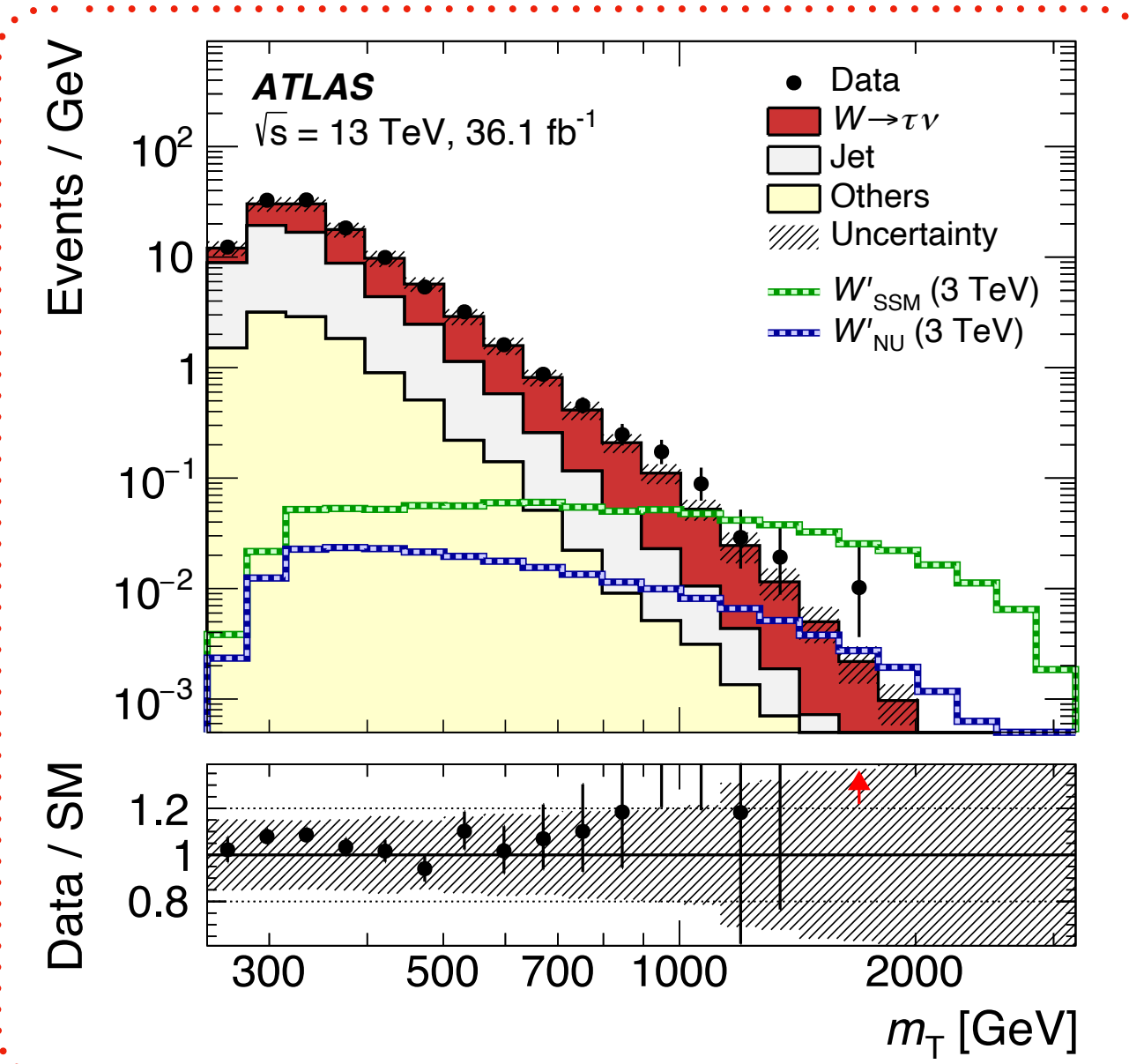
Read the reconstructed events from the simulation output and apply experiment-specific selections.

##### ATLAS selection

- Lepton veto: no electrons or muons
- $\geq 1$  hadronic tau with  $p_T > 80$  GeV,  $|\eta| < 2.3$
- $E_T^{\text{miss}} > 150$  GeV
- $m_T > 250$  GeV, with  $m_T = \sqrt{2p_T^\tau E_T^{\text{miss}}(1 - \cos \Delta\phi)}$

##### CMS selection

- Lepton veto: no electrons or muons
- $\geq 1$  hadronic tau with  $p_T > 80$  GeV,  $|\eta| < 2.1$
- $E_T^{\text{miss}} > 200$  GeV



# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## 1. Target

## 3. Collider Simulation

## 4 Numerical Analysis

### step 2: signal template construction

For events passing selection, histogram  $m_T$  into ATLAS or CMS bins.

$$s_i^{(g=1)} = \frac{N_i^{\text{pass}}}{N_{\text{gen}}} \times \sigma(g=1) \times \mathcal{L}$$

where

- $N_{\text{gen}}$  is the number of reconstructed events
- $\sigma(g=1)$  is the cross section at coupling  $g=1$
- $\mathcal{L}$  is the luminosity ( $36.1 \text{ fb}^{-1}$  ATLAS,  $35.9 \text{ fb}^{-1}$  CMS)

Signal scaling:  $s_i(g) = g^4 s_i^{(g=1)}$

### 3.3 Profile likelihood analysis

For each bin,

$$-\ln L_i(\theta_i) = -[n_i \ln \mu_i - \mu_i] + \frac{1}{2} \frac{\theta_i^2}{(\delta_i/b_i)^2}$$

where

- $n_i$  observed events
- $b_i$  expected SM background
- $\delta_i$  systematic uncertainty
- $\theta_i$  nuisance parameter
- $s_i(g) = g^4 s_i^{(g=1)}$
- $\mu_i = b_i(1 + \theta_i) + s_i$

### Procedure to find the $2\sigma$ exclusion contour

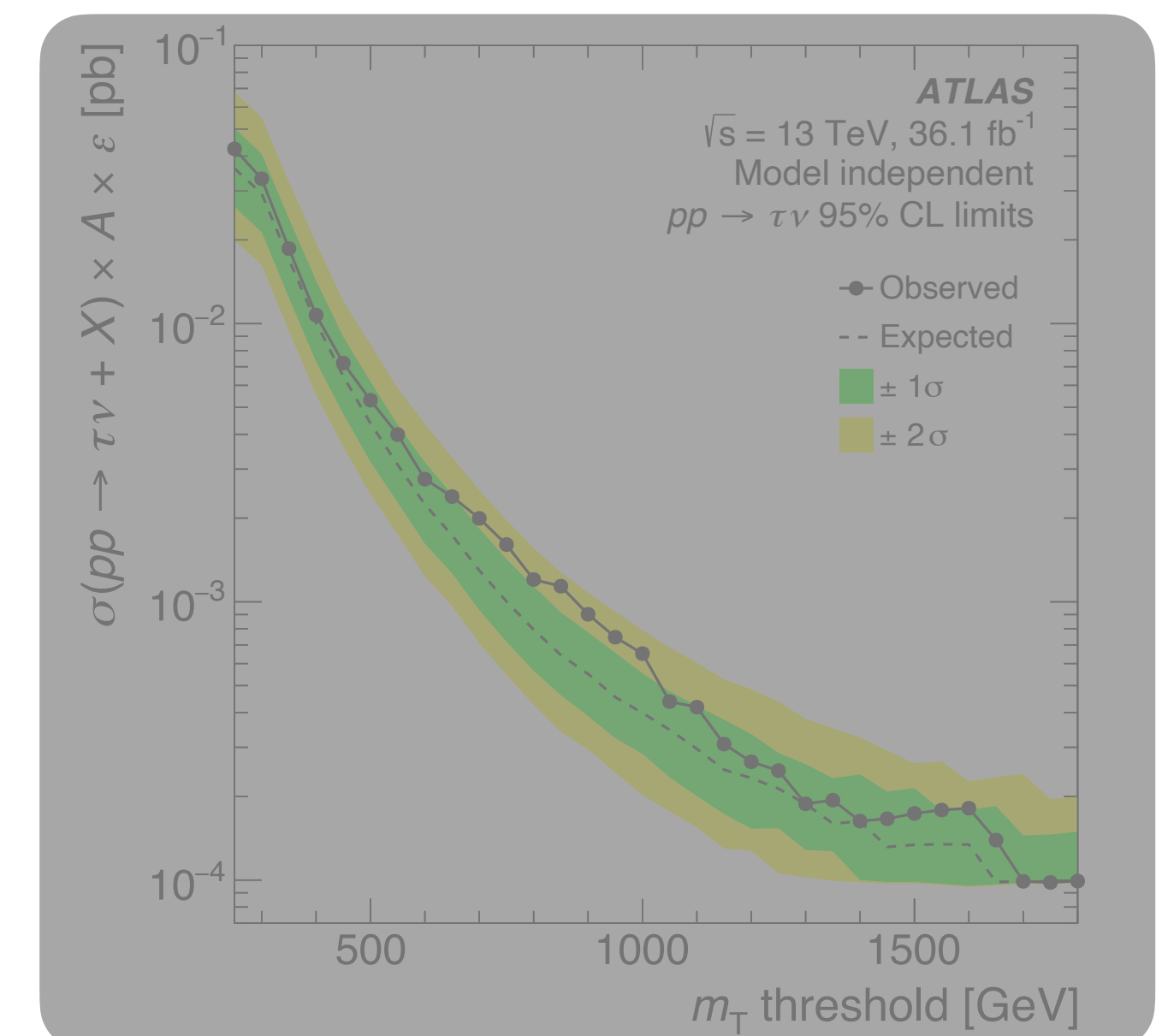
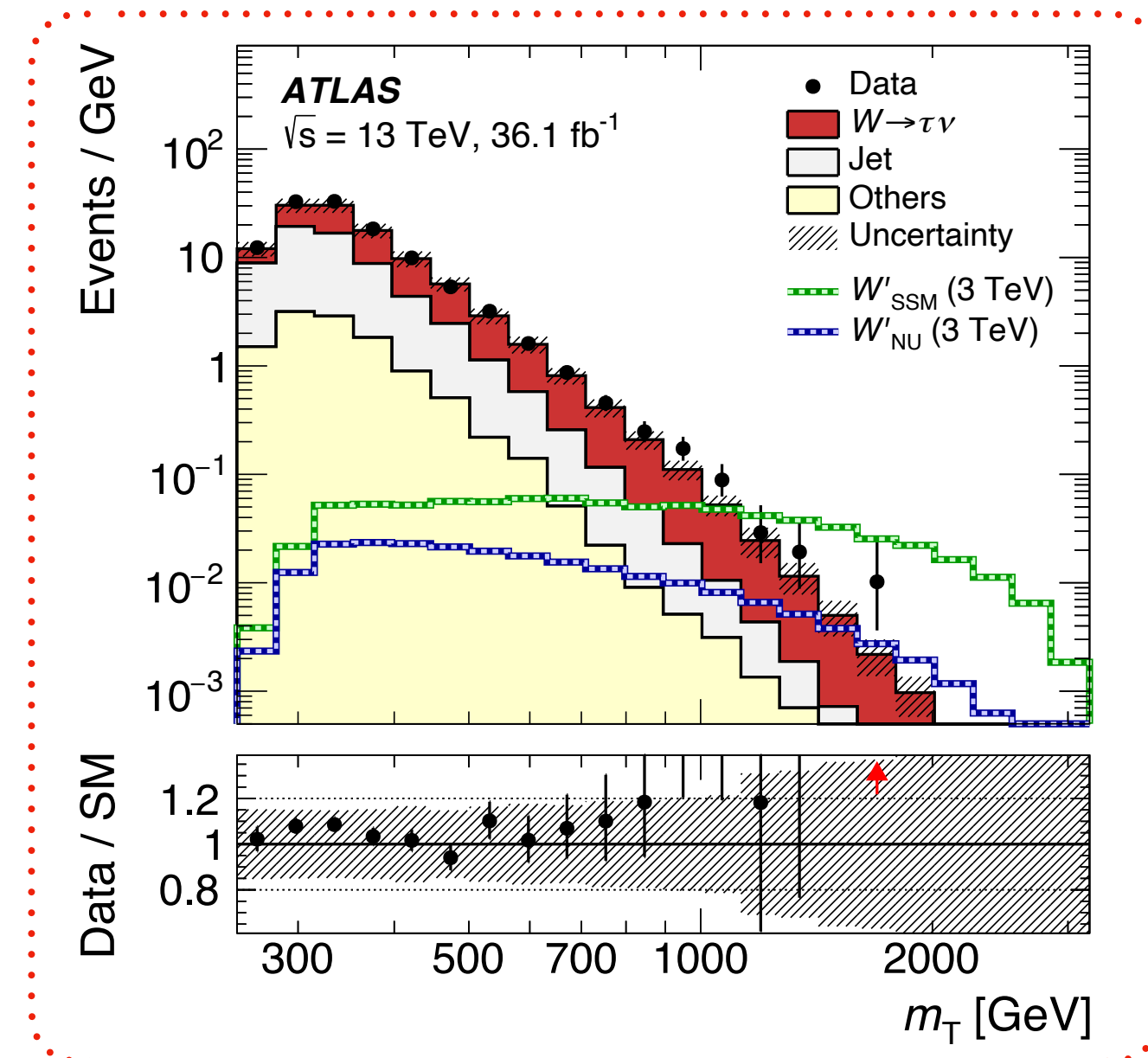
**step 1 profiling:** Minimize  $-\ln L_i(\theta_i)$  numerically with constraint  $\mu_i > 0$ .

**step 2 combine ATLAS + CMS Likelihood:**  $\ln \mathcal{L}_{\text{comb}} = \ln \mathcal{L}_{\text{ATLAS}} + \ln \mathcal{L}_{\text{CMS}}$

**step 3 extract exclusion region:**

1. Find best-fit  $\hat{g}$
2. Find the exclusion curve by considering  $g_{\text{excl}}$  satisfying  $-2[\ln \mathcal{L}(g_{\text{excl}}) - \ln \mathcal{L}(\hat{g})] = 4$

### 3.4 Plot Figure





# 3. Constraints on $U_1$ LQ from LHC mono- $\tau$ search

## 1. Target

## 3. Collider Simulation

## 4 Numerical Analysis

### step 2: signal template construction

For events passing selection, histogram  $m_T$  into ATLAS or CMS bins.

$$s_i^{(g=1)} = \frac{N_i^{\text{pass}}}{N_{\text{gen}}} \times \sigma(g=1) \times \mathcal{L}$$

where

- $N_{\text{gen}}$  is the number of reconstructed events
- $\sigma(g=1)$  is the cross section at coupling  $g=1$
- $\mathcal{L}$  is the luminosity ( $36.1 \text{ fb}^{-1}$  ATLAS,  $35.9 \text{ fb}^{-1}$  CMS)

Signal scaling:  $s_i(g) = g^4 s_i^{(g=1)}$

### 3.3 Profile likelihood analysis

For each bin,

$$-\ln L_i(\theta_i) = -[n_i \ln \mu_i - \mu_i] + \frac{1}{2} \frac{\theta_i^2}{(\delta_i/b_i)^2}$$

where

- $n_i$  observed events
- $b_i$  expected SM background
- $\delta_i$  systematic uncertainty
- $\theta_i$  nuisance parameter
- $s_i(g) = g^4 s_i^{(g=1)}$
- $\mu_i = b_i(1 + \theta_i) + s_i$

### Procedure to find the $2\sigma$ exclusion contour

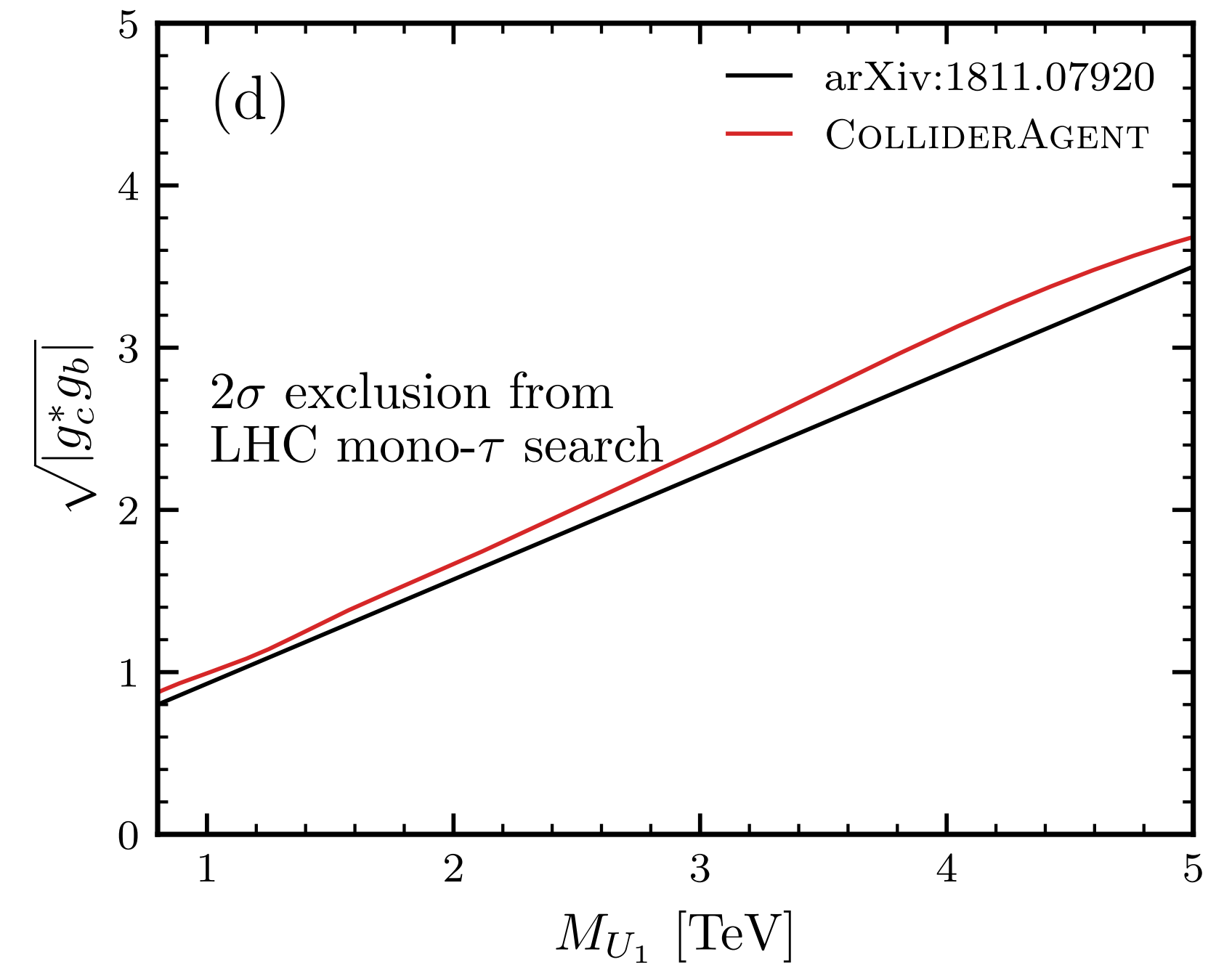
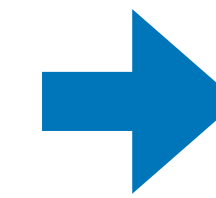
**step 1 profiling:** Minimize  $-\ln L_i(\theta_i)$  numerically with constraint  $\mu_i > 0$ .

**step 2 combine ATLAS + CMS Likelihood:**  $\ln \mathcal{L}_{\text{comb}} = \ln \mathcal{L}_{\text{ATLAS}} + \ln \mathcal{L}_{\text{CMS}}$

**step 3 extract exclusion region:**

1. Find best-fit  $\hat{g}$
2. Find the exclusion curve by considering  $g_{\text{excl}}$  satisfying  $-2[\ln \mathcal{L}(g_{\text{excl}}) - \ln \mathcal{L}(\hat{g})] = 4$

## 3.4 Plot Figure



# 4. Parameter scan in a minimal $U(1)_{B-L}$ model



PUBLISHED FOR SISSA BY SPRINGER

RECEIVED: May 14, 2016

ACCEPTED: July 1, 2016

PUBLISHED: July 18, 2016

## $Z'$ , Higgses and heavy neutrinos in $U(1)'$ models: from the LHC to the GUT scale

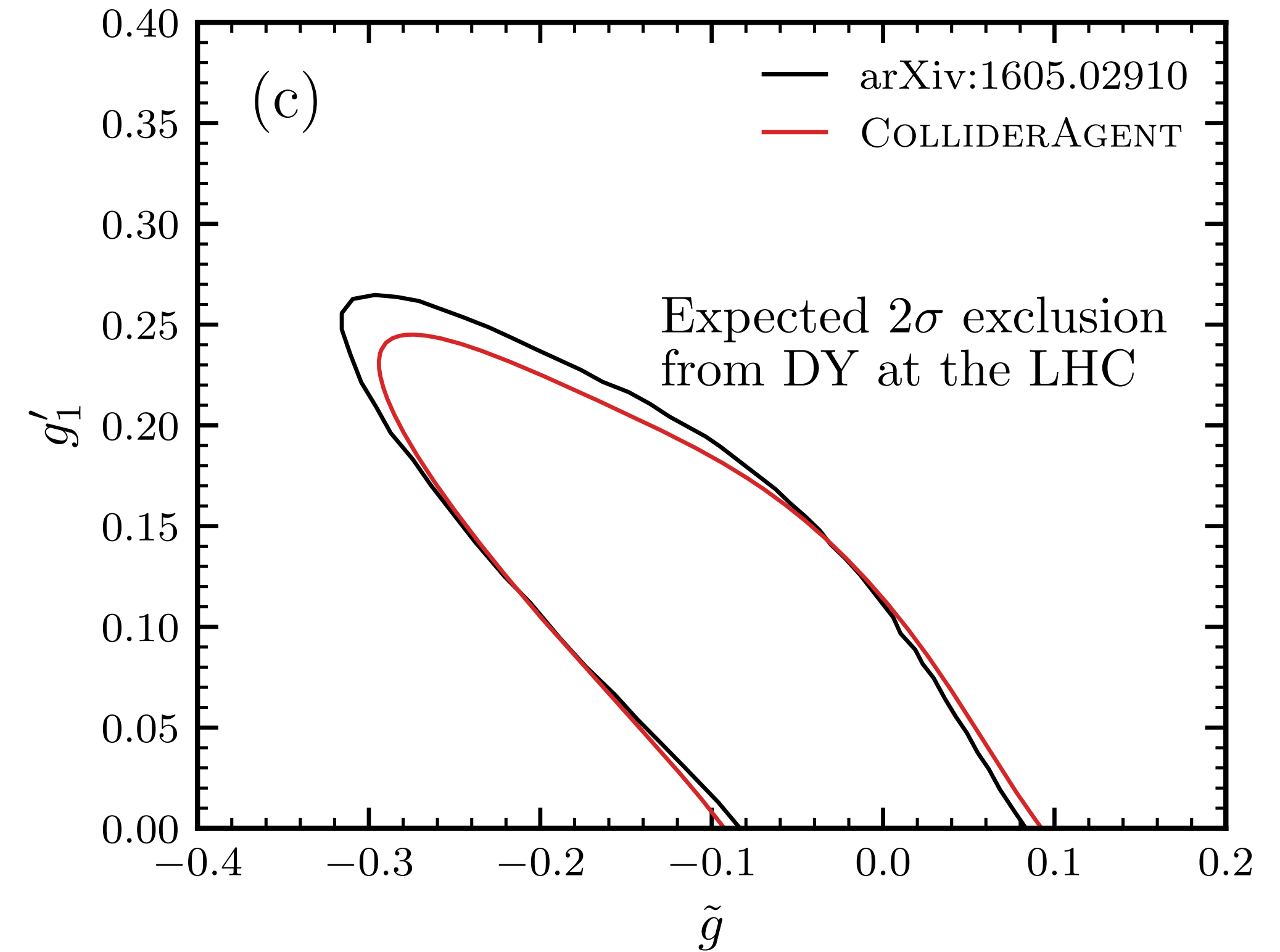
Elena Accomando,<sup>a</sup> Claudio Corianò,<sup>b,c</sup> Luigi Delle Rose,<sup>a</sup> Juri Fiaschi,<sup>a</sup> Carlo Marzo<sup>c</sup>  
and Stefano Moretti<sup>a</sup>

### Part of the Lagrangian

$$\mathcal{L}_{Z'ff} = \sum_f Z'_\mu \bar{f} \gamma^\mu (C_{f,L} P_L + C_{f,R} P_R) f$$

$$C_{f,L} = -e \frac{\sin \theta'}{\sin \theta_W \cos \theta_W} (T_f^3 - \sin^2 \theta_W Q_f) + (\tilde{g} Y_{f,L} + g'_1 z_f) \cos \theta'$$

$$C_{f,R} = e \frac{\sin \theta_W \sin \theta'}{\cos \theta_W} Q_f + (\tilde{g} Y_{f,R} + g'_1 z_f) \cos \theta'$$



Parameter Scan



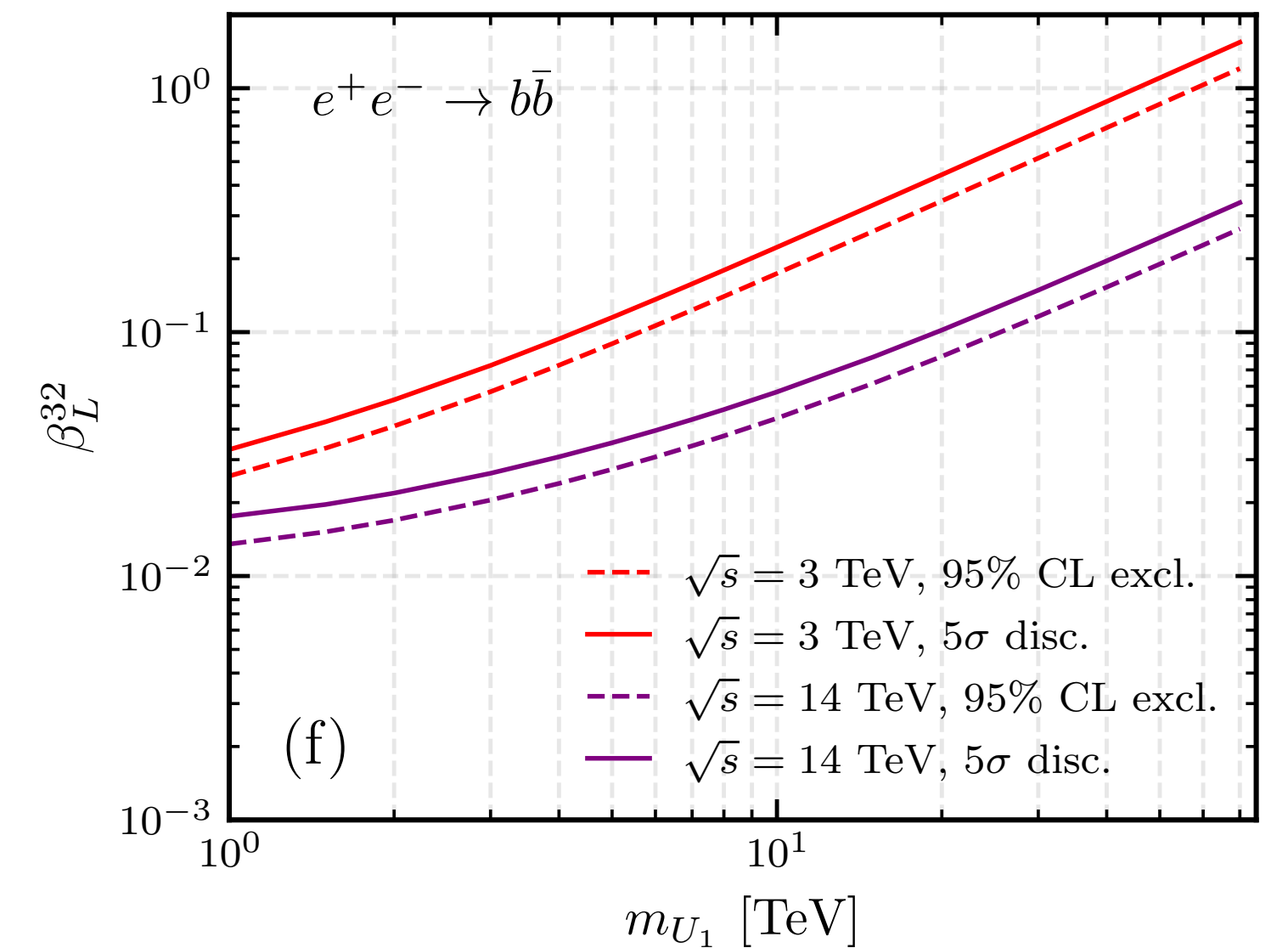
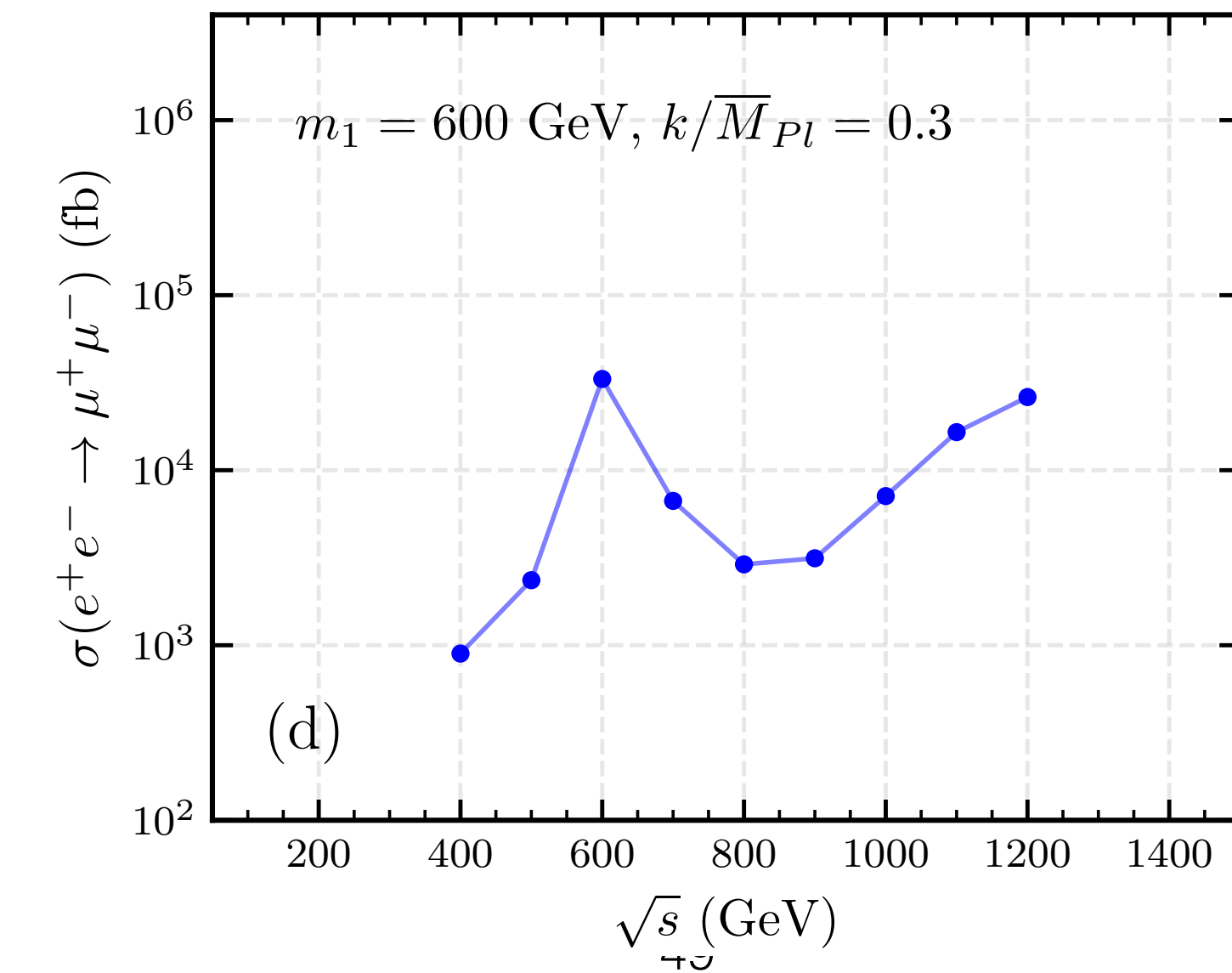
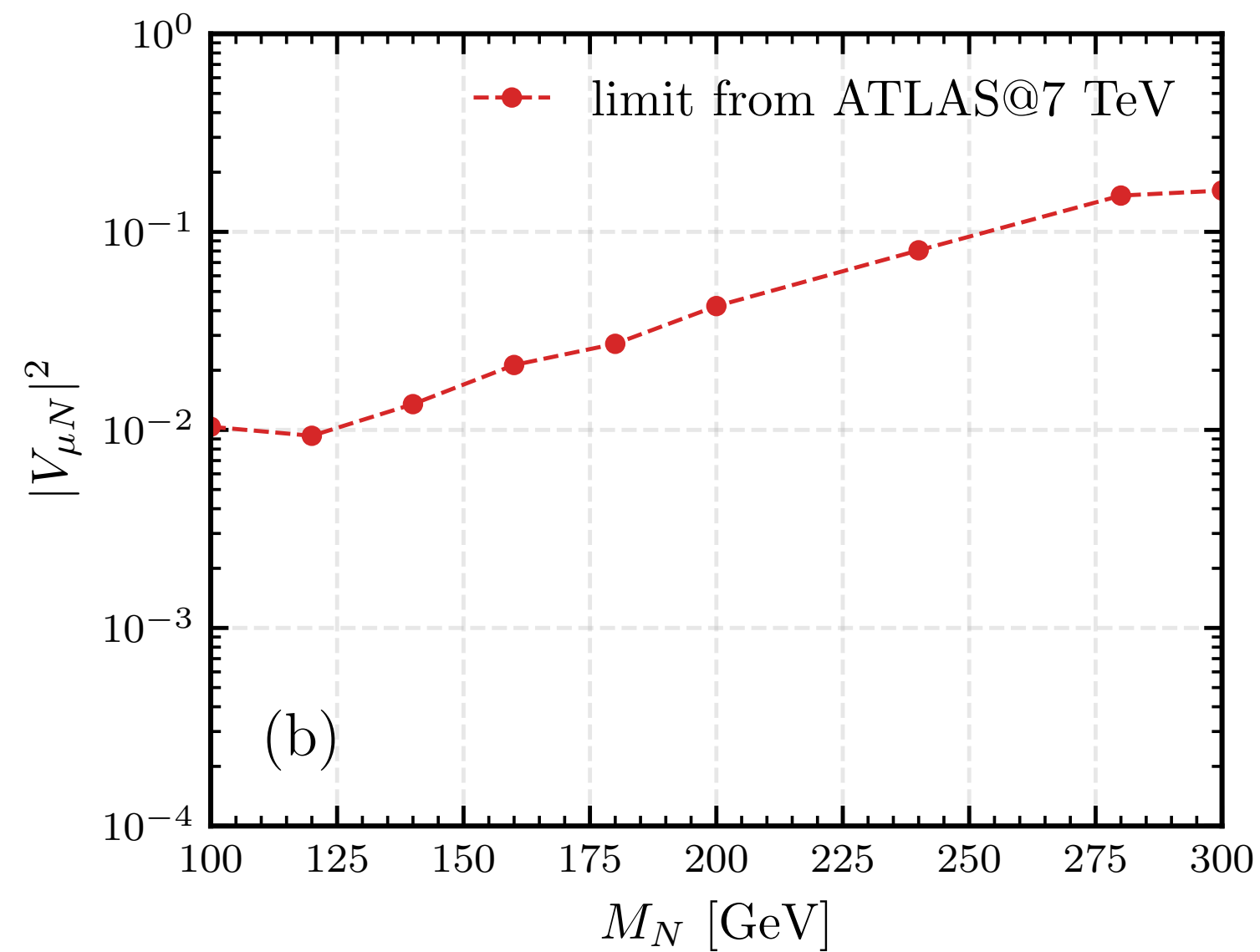
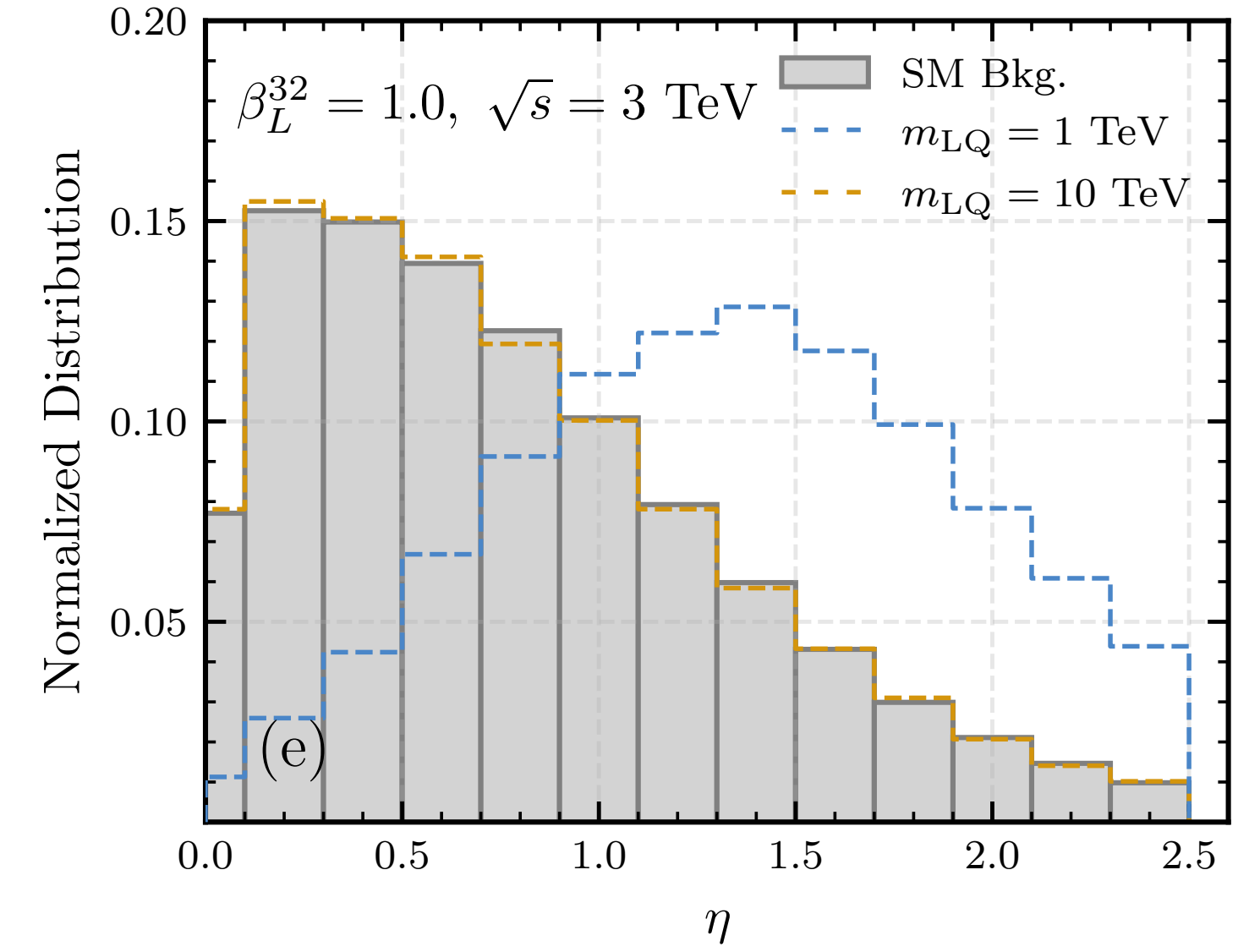
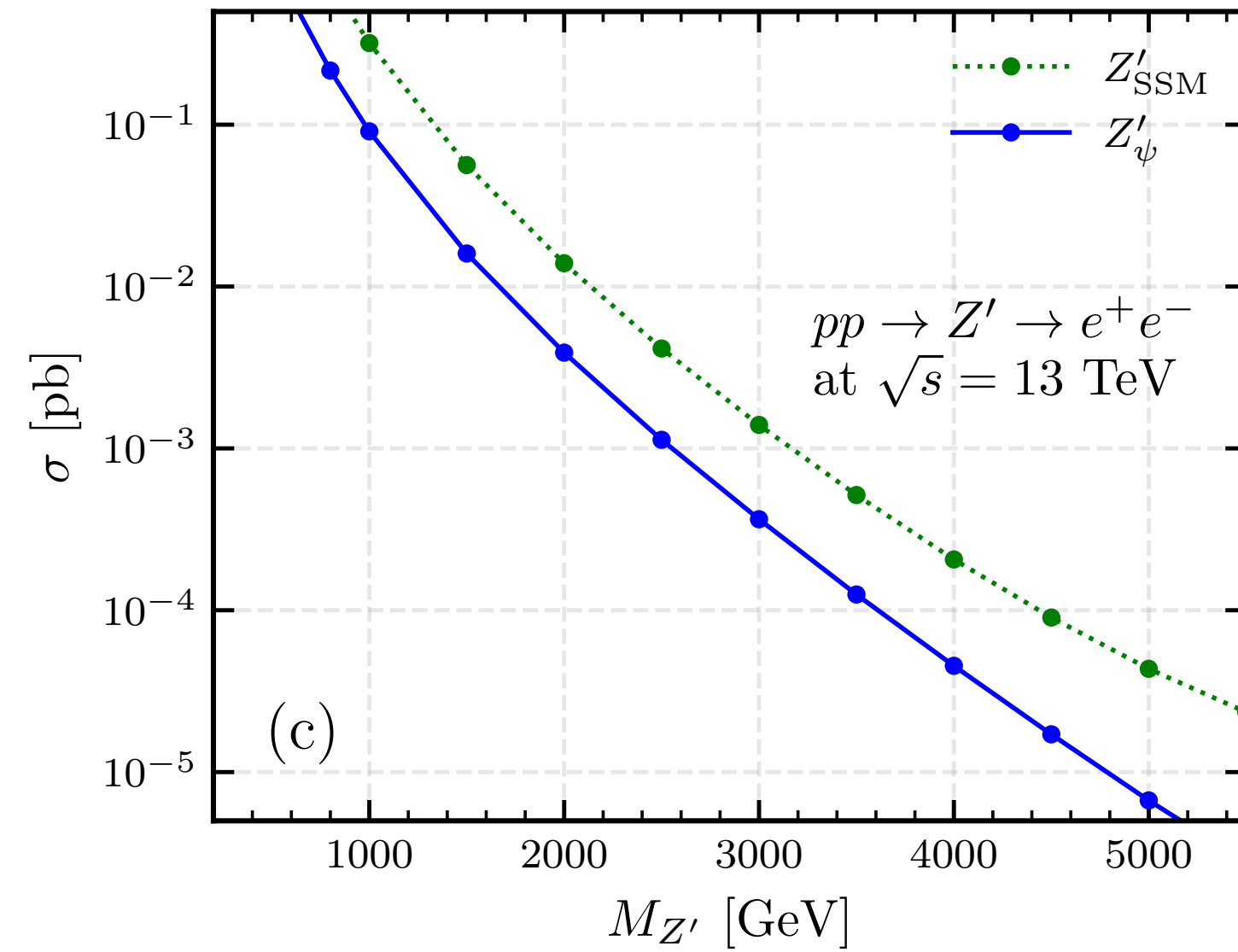
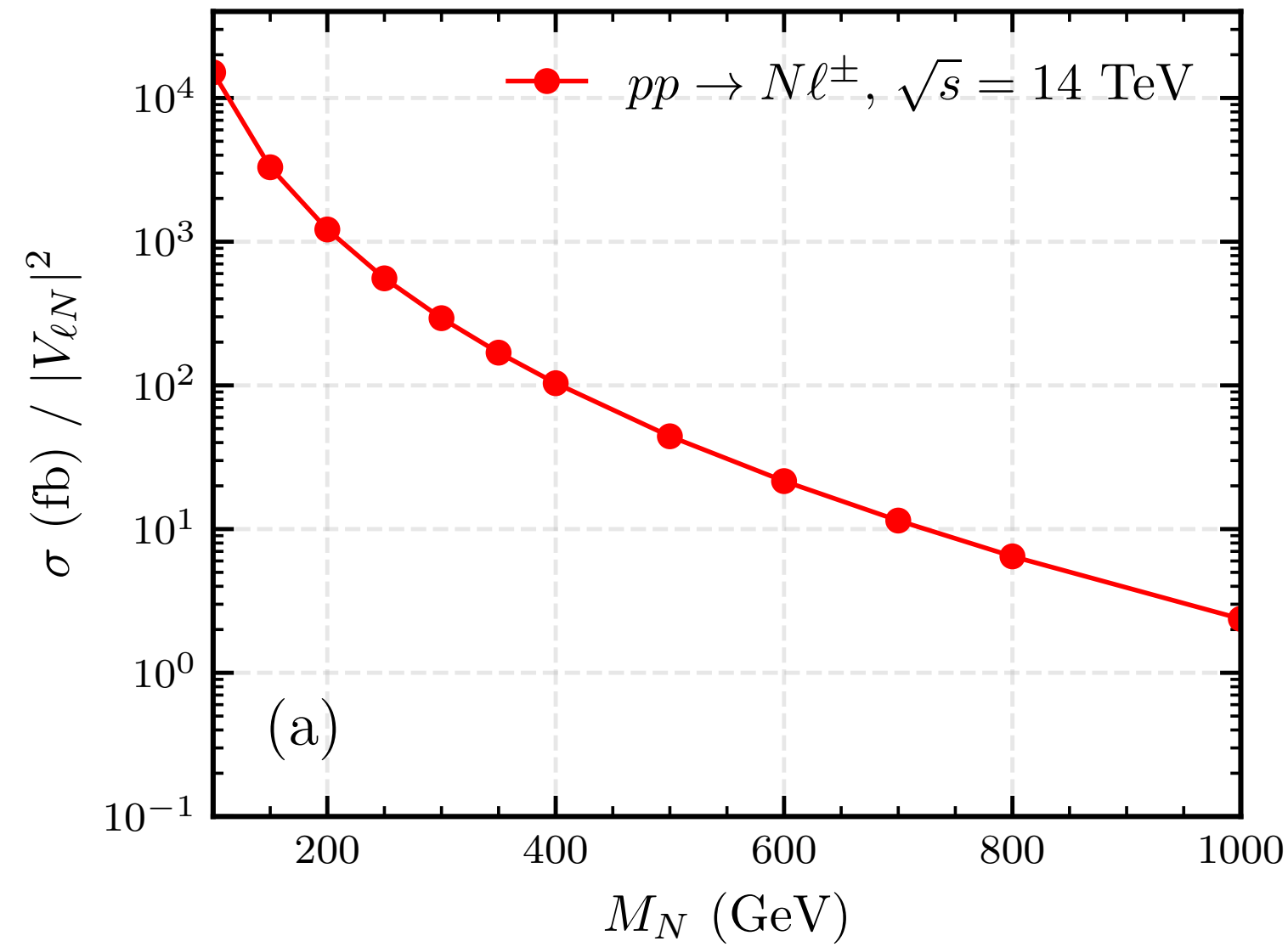
# Other examples

(a, b) P. S. B. Dev, A. Pilaftsis, and U.-k. Yang, arXiv: 1308.2209

(c) CMS, arXiv: 2103.02708

(d) H. Davoudiasl, J. L. Hewett, and T. G. Rizzo, hep-ph/9909255

(e, f) P. Asadi, R. Capdevilla, C. Cesarotti, and S. Homiller, arXiv: 2104.05720



# 我们这套架构还能做什么？

- ▶ 暗物质: `microOMEGAs`, `DDCalc`, `DRAKE`, .....
- ▶ 味物理: `MatchETE`, `MatchMakerEFT`, `DsixTools`, `Flavio`, `EOS`, .....
- ▶ 圈图计算: `QGraf`, `FIRE`, `KIRA`, `AMFlow`, `Libra`, .....
- ▶ 引力波: `ELENA`, `CosmoTransitions`, `BSMPT`, .....

**以及更一般的物理领域！**

如果有好的想法也欢迎联系我们



# 对撞机物理方面的其它相关研究

- ▶ **Tony Menzo, et al, 2512.15867**
- ▶ **Tilman Plehn, Daniel Schiller, Nikita Schmal, 2601.21015**
- ▶ **Prateek Agrawal, Nathaniel Craig, Amalia Madden, Inigo Valenzuela Lombera, 2603.22538**
- ▶ **parallel developments for agent for experiments**

# 总结

- ▶ ColliderAgent是首个仅依赖自然语言指令，即可从 Lagrangian 出发，自动推进至完整唯象学分析结果的自动化系统
- ▶ ColliderAgent成功实现了对 8 篇经典论文的端到端复现，涉及 Leptoquark、类轴子、有效场论，以及从部分子水平到探测器水平的全流程模拟与参数扫描
- ▶ ColliderAgent背后的智能体架构设计（包括多智能体、Skills 和通用计算平台 Magnus）在其他物理领域也有巨大的应用潜力

希望本工作提出的智能体架构可以推动一种新的科研分工方式：  
科学家负责idea、价值判断等；AI负责干活！

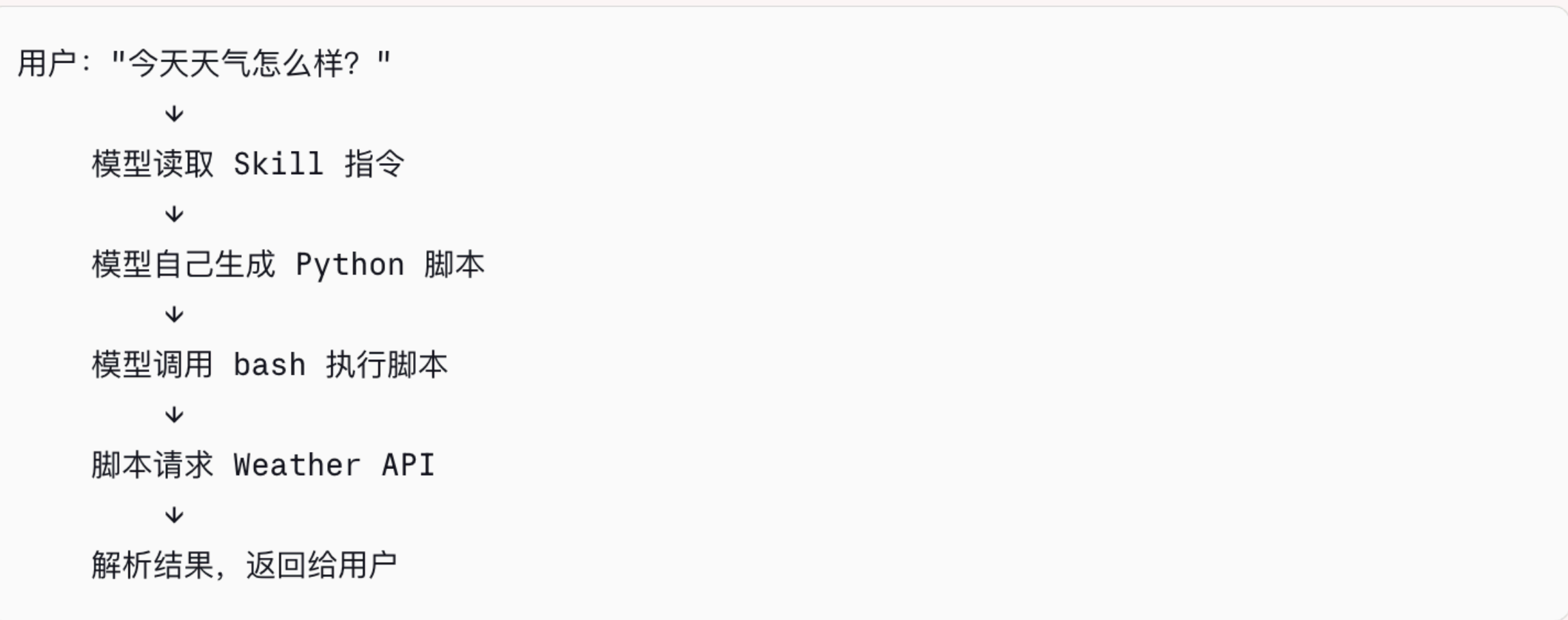
谢谢大家！



# Backup

# Skills vs. MCP

## Skills 方式：模型是执行者



模型亲自下场写代码、跑代码——**模型是执行链条的一部分。**

## MCP 方式：外部 Server 是执行者



模型只是**发出指令**，脏活累活都在 MCP Server 里完成了。

	Skills	MCP
执行者	模型本身（读指令、生成代码、自己跑）	外部 Server（模型只发请求）
本质	给模型看的说明书	模型可调用的外部服务
标准化程度	✅ 开放标准，跨工具通用	✅ 开放标准，跨工具通用