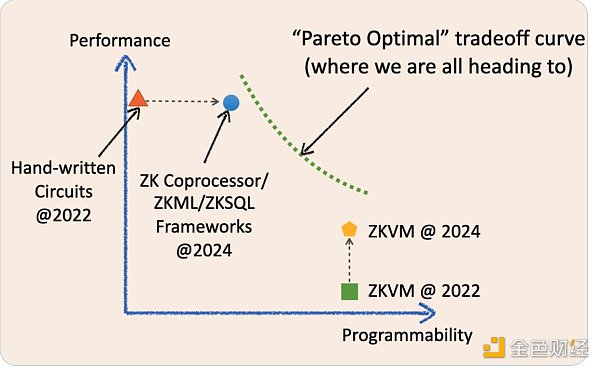
专用 VS 通用 ZK：哪一个才是未来？

作者：mo 来源：X，@no89thkey 翻译：善欧巴，本站

让我尝试用一个数字来回答这个问题：



我们是否有可能仅收敛于权衡平面中的一个神奇的最佳点？不，链下可验证计算的未来是一条连续曲线，模糊了专用 ZK 和通用 ZK 之间的界限。请允许我解释一下这些术语在历史上是如何演变的，以及它们在未来将如何融合。

两年前，“专业”的 ZK 基础设施意味着 circom、Halo2 和 arkworks 等低级电路框架。用这些构建的 ZK 应用程序本质上是手写的 ZK 电路。它们对于非常具体的任务来说速度快且成本低，但通常难以开发和维护。它们类似于当今IC行业中的各种专用集成电路芯片（物理硅），例如 NAND 芯片和控制器芯片。

然而，在过去的两年里，“专业化”的 ZK 基础设施已经发展成为更加“通用化”的基础设施。我们现在拥有 ZKML、ZK 协处理器和 ZKSQL 框架，它们提供易于使用和高度可编程的 SDK，可以构建不同类别的 ZK 应用程序，而无需编写一行 ZK 电路代码。例如，ZK 协处理器允许智能合约无需信任地访问历史区块链状态/事件/交易，并对这些数据运行任意计算。ZKML 使智能合约能够可靠地利用人工智能推理结果来实现广泛的机器学习模型。

这些演进的框架显著提高了其目标领域内的可编程性，同时仍然保持高性能和低成本，因为抽象层（SDK/API）很薄并且接近裸机电路。它们类似于 IC 市场中的 GPU、TPU 和 FPGA：它们是可编程的领域专家。

ZKVM 在过去两年中也取得了长足的发展。值得注意的是，所有通用 ZKVM 都是构建在低级、专用 ZK 框架之上的。这个想法是，您可以用高级语言（甚至比 SDK/API 更加用户友好）编写 ZK 应用程序，这些应用程序可以编译为指令集（RISC-V 或 WASM 类）的专用电路的组合。在我们对 IC 行业的类比中，它们就像 CPU 芯片。ZKVM 是低级 ZK 框架之上的一个抽象层，就像 ZK 协处理器等一样，尽管是一个更厚的层。

正如一位智者曾经说过的，一层抽象可以解决每个计算机科学问题，但同时会产生另一个问题。我的朋友，权衡就是这里游戏的名称。从根本上说，对于 ZKVM，我们在性能和通用性之间进行权衡。

两年前，ZKVM 的“裸机”性能确实很糟糕。然而，短短两年时间，ZKVM 的性能有了显着提升。为什么？因为这些“通用”的 ZKVM 已经变得更加“专业”了！性能提升的一个关键领域来自“预编译”。这些预编译是专门的 ZK 电路，可以计算常用的高级程序，例如 SHA2 和各种签名验证，比将它们分解为指令电路的正常流程要快得多。

因此，现在的趋势已经很明显了。专业化的 ZK 基础设施正变得越来越通用，而通用化的 ZKVM 也正变得更加专业化！

对于过去几年的这两种解决方案，优化是实现比以前更好的权衡点：在一个点上做得更好，而不牺牲另一点。这就是为什么双方都觉得“我们绝对是未来”。

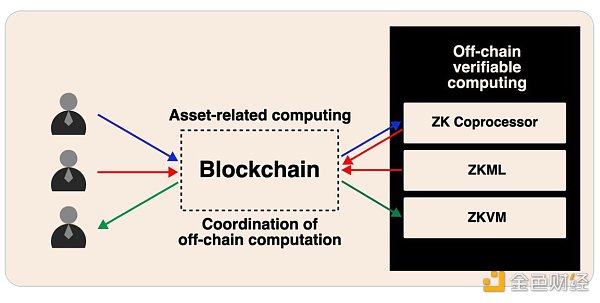
然而，计算机科学智慧告诉我们所有人，在某一时刻，我们将遇到“帕累托最优墙”（绿色虚线），在这种情况下，我们无法在不牺牲另一种特征的情况下改善一种特征。因此，一个价值百万美元的问题出现了：其中一种会在适当的时候完全取代另一种吗？

如果 IC 行业的类比能有所帮助的话：CPU 的市场规模为 1,260 亿美元，整个 IC 行业，加上所有“专用”IC，市场规模为 5,150 亿美元。我确实相信，在微观层面上，历史将在这里押韵，并且不会相互取代。

话虽如此，今天没有人说，“嘿，我正在使用一台完全由通用 CPU 驱动的计算机”，或者“嘿，看看这个由专用 IC 驱动的奇特机器人。”是的，我们确实应该从宏观层面来看待这件事，未来就是提供一个权衡曲线，让开发者根据个人需求灵活选择。

未来，领域专家 ZK 基础设施和通用 ZKVM 可以并且将会协同工作。这可以多种形式发生。如今，最简单的方法已经成为可能。例如，您可能使用 ZK 协处理器在区块链交易的长期历史中生成一些计算结果，但这些数据之上的计算业务逻辑非常复杂，以至于您无法轻松地在 SDK/API 中表达它。

你能做的就是获得数据和中间计算结果的高性能和低成本的 ZK 证明，然后通过证明递归将它们汇集到广义 VM。



虽然我确实认为这些类型的辩论很有趣，但我知道我们都在为由链下可验证计算提供支持的区块链构建异步计算的未来。当我们看到未来几年出现大规模用户采用的用例时，我相信这场争论可以轻松解决。