**Классы сложности задач**

В теории алгоритмов **классами сложности** (КС) называются множества вычислительных задач, примерно одинаковых по сложности вычисления. Если точнее, то КС — это множество предикатов, использующих для вычисления примерно одинаковый объем ресурсов. В качестве ресурсов обычно берется **время исполнения** или используемый при исполнении **объем памяти**.

Для каждого класса существует категория задач, которые являются «самыми сложными» - **полными**. Это означает, что любая задача из класса сводится к такой задаче, и притом сама задача лежит в классе. Полные задачи — хороший инструмент для доказательства равенства классов: достаточно для одной такой задачи доказать ее принадлежность более низкому классу, и равенство данных классов будет доказано.

Основные классы сложности:

* **EXPSPACE/NEXPSPACE** - задачи, которые могут быть разрешены детерминированной/недетерминированной машиной Тьюринга с экспоненциальным ограничением пространства; доказано, что EXPSPACE = NEXPSPACE (теорема Савича);
* **NEXPTIME** - задачи, разрешимые недетерминированной машиной Тьюринга за экспоненциальное время;
* **EXPTIME** - задачи, разрешимые детерминированной машиной Тьюринга за экспоненциальное время;
* **PSPACE/NPSPACE** – задачи, разрешимые детерминированной/недетерминированной машиной Тьюринга с полиномиальным ограничением пространства; доказано, что PSPACE = NPSPACE (теорема Савича);  
  **PH** – обобщение NP, получаемое квантифицированием NP-задач; иначе это объединение всех классов сложности из полиномиальной иерархии; полиномиальная иерархия обобщает классы P, NP и co-NP до вычислений с оракулом (вычисление на машине Тьюринга с некоторым устройством, способным угадать решение проблемы разрешимости за один такт машины);
* **NP** – задачи, разрешимые на недетерминированной машине Тьюринга за полиномиальное время;
* **P** – задачи, разрешимые на детерминированной машине Тьюринга за полиномиальное время;
* **BQP** –задачи, решаемые вероятностными алгоритмами на квантовом компьютере за полиномиальное время; доказано, что ;
* **BPP**  - задачи, решаемые с использованием вероятностных алгоритмов за полиномиальное время; доказано, что .

Вложенность классов сложности друг в друга имеет вид:

**Блокчейн**

**ZK-STARK**

**Аннотация**

Человеческое достоинство требует, чтобы личная информация, например, медицинские и судебные сведения, была скрыта от общественности. Но завесой секретности, предназначенной для сохранения конфиденциальности, также могут злоупотреблять для сокрытия лжи и обмана учреждения, которым доверены конфиденциальные данные, несправедливо нанося ущерб гражданам и подрывая доверие к подобным организациям.

Системы доказательства нулевого знания (ZK) являются оригинальным криптографическим решением разногласий между принципами персональной конфиденциальности и институциональной целостности, обеспечивая соблюдение последних и не нанося ущерб первому. Общественное доверие требует прозрачности от систем ZK, что подразумевает функционирование без опоры на какую-либо доверенную сторону и отсутствие уязвимостей, которые могли бы быть использованы влиятельными сторонами для фальсификации. Для систем ZK, работающих с большими данными, крайне важно, чтобы процесс публичной проверки масштабировался сублинейно относительно размера данных. Прозрачные доказательства ZK, которые могут быть проверены за время, экспоненциально меньшее размера данных, были впервые описаны в 1990-х годах, но ранние версии были непрактичны, и до сих пор ни одна ZK система, реализованная в коде (в том числе используемая криптовалютами, такими как Zcash), не обеспечивает одновременно прозрачность и экспоненциальное ускорение проверки для общих вычислений.

В данной работе приводится первая реализация прозрачной системы ZK (ZK-STARK), в которой проверка масштабируется экспоненциально быстрее, чем размер базы данных, и, кроме того, это экспоненциальное ускорение проверки наблюдается конкретно для значимых и последовательных вычислений, описанных далее. Описываемая система использует несколько последних достижений в области **интерактивных доказательств с оракулом** (IOP), таких как “быстрая” (за линейное время) IOP система для кодов корректировки ошибок.

Данная система проверки концепции позволяет полиции доказать общественности, что профиль ДНК кандидата в президенты не фигурирует в базе данных криминалистических профилей ДНК, которую ведет полиция. Доказательство, которое генерируется полицией, не зависит от какой-либо внешней доверенной стороны и не раскрывает никакой дополнительной информации о содержимом БД или о профиле кандидата. В частности, никакая информация ДНК профиля не раскрывается никакой стороне за пределами полиции. Доказательство короче, чем размер БД ДНК, и верифицируется быстрее, происходит наивный перебор данных в БД.

**Введение**

**Масштабируемая проверка целостности вычислений на конфиденциальных данных**

Проблема, рассматриваемая здесь, лучше всего иллюстрируется гипотетическим примером: предположим, что полиция (P), отвечающая за национальную БД судебной экспертизы профилей ДНК (D), утверждает, что профиль ДНК (p) кандидата в президенты, который вскоре будет назначен и предположительно будет коррумпирован, не фигурирует в D. Могут ли криптографические протоколы убедить сомневающуюся общественность поверить в это утверждение без ущерба для D или p, не полагаясь на какую-либо внешнюю доверенную сторону (например, главный судья), и с “разумными” вычислительными ресурсами?

Пример соответствия профиля ДНК (DPM) является частным случаем более общей проблемы. Произвольная сторона (P), выполняющая вычисления (C) на наборе данных (D), может иметь стимул подделывать правильный выход (С(D)), вызывая проблему вычислительных целостности (CI) — гарантировать, что P действительно предоставила C(D) а не результат, по тем или иным причинам более благоприятный для P. Когда набор данных D является общедоступным, любая сторона (V), заинтересованная в проверке CI, может наивно выполнить С на D и сравнить результат с полученным от P. Однако такое решение не масштабируется, потому что время, потраченное верификатором (ТV) также велико, как и время, необходимое для выполнения программы (ТC) и V должен прочитать весь датасет D. Схемы обязательств, основанные на криптографических хеш-функциях, широко используются для вычисления нескольких неизменяемых “отпечатков” cmt для состояния в момент времени t на большом наборе данных Dt. Обычно cmt имеет незначительную длину по сравнению с Dt и может быть легко размещен в блокчейне как публичное уведомление. Таким образом, искомое решение CI должно иметь масштабируемую проверку, в которой время проверки и сложность связи масштабируются примерно так же, как log TC и |cmt| (битовая длина cmt), а не как TC и |Dt|; по крайней мере, время проверки/связи должно быть строго меньше, чем TC и |Dt|.

Когда набор данных D содержит конфиденциальную информацию, наивное решение больше не может быть реализовано, и сторона P, ответственная за D, может скрывать нарушения вычислительной целостности под предлогом секретности. Доминирующие методы обеспечения CI над конфиденциальными данными полагаются на “доверенную сторону”, например, аудитора или бухгалтера, которая наивно проверяет расчеты от имени общественности. Это решение по-прежнему не обеспечивает масштабирования, как и в случае публичных данных, и, что еще хуже, требует, чтобы общественность доверяла третьей стороне, что создает потенциальную единственную точку сбоя в протоколе, поскольку эта третья сторона — в той мере, в какой это может быть согласовано — может быть скомпрометирована злоумышленниками.

**Системы доказательств и аргументов с нулевым разглашением** представляют собой автоматизированные протоколы, заменяющие реальных аудиторов в качестве средств обеспечения вычислительной целостности на конфиденциальных данных для любых эффективных вычислений, устраняя коррумпированность и снижая вычислительные затраты. ZK-система S для вычислений C представляет собой пару вероятностных алгоритмов S = (P, V):

* P – алгоритм доказательства вычислительной целостности;
* V – алгоритм верификации доказательства P.

Полнота и обоснованность S подразумевают, что P может эффективно доказать все истинные утверждения, но не сможет убедить V в каких-либо ложных (кроме, быть может, ничтожной вероятности). Самые первые теоретические разработки систем ZK с масштабируемыми верификаторами для общих вычислений, обсуждавшиеся в начале 1990-х годов, были основаны на вероятностно проверяемых доказательствах (PCP). Знаменитая теорема PCP предложила удивительный компромисс между временем выполнения, затраченным доказывающим на построение доказательства (TP), и временем выполнения, затраченным проверяющим, верифицирующим это доказательство (TV). Этот компромисс означает, что время доказательства увеличивается полиномиально по сравнению с наивным временем вычисления (TP = TСO(1)), в то время как время проверки уменьшается экспоненциально по отношению к нему (TV = logO(1) TC).

Система ZK, основанная на теореме PCP (ZK-PCP), имеет три дополнительных преимущества, которые необходимы для обеспечения доверия к целостности вычислений. Во — первых, предположения, на которых основана безопасность этого подхода - существование устойчивых к коллизиям хэш-функций для интерактивных решений и общий доступ к случайной функции (“модель случайного оракула”) для неинтерактивных — делают данное решение устойчивым к атакам крупномасштабных квантовых компьютеров (постквантово безопасные). Ожидаемое увеличение масштабов квантовых компьютеров и призыв к постквантовым криптографическим протоколам, например, Национальным институтом стандартов и Технологии (NIST), подчеркивают важность постквантового безопасного решения ZK.

Во-вторых, ZK-PCP являются системами доказательства знаний (proof of knowledge - POK) или, в случае описанной выше реализации, системами аргументации знаний (argument of knowledge - ARK). Неформально, в контексте примера DPM, ZK-ARK является доказательством того, что полиция использовала “истинный” набор данных Dt и профиль ДНК p кандидата в президенты.

В-третьих, что наиболее важно, ZK-PCP прозрачны (“публичная случайность”), т.е. случайность, используемая верификатором, является общедоступной; в частности, настройка ZK-PCP не требует внешнего доверенного этапа настройки, в отличие от более новых решений ZK, в том числе используемых криптовалютой Zcash. Прозрачность необходима для постоянного общественного доверия, поскольку она серьезно ограничивает способность даже самых могущественных сторон P злоупотреблять системой, и, таким образом, прозрачные системы - это те, которым общественность может надежно доверять, пока в наблюдаемой вселенной существует что-либо непредсказуемое (источники случайности).

Подводя итог, можно сказать, что ZK-PCP являются отличным методом обеспечения общественного доверия к IC на конфиденциальных данных и имеют шесть основных достоинств:

* прозрачность;
* универсальность — распространяются на любые эффективные вычисления С, даже если они требуют дополнительного (а возможно, и конфиденциальную) входа типа Dt;
* конфиденциальность (ZK) — не компрометируют вспомогательные входы типа Dt
* пост-квантовая безопасность;
* доказательство/аргумент знания;
* масштабируемость проверки.

Хотя ZK-PCP известны с середины 1990-х годов, до сих пор ни один из них не был реализован в коде, потому что, согласно недавнему исследованию [110], “доказательства, вытекающие из теоремы PCP (несмотря на асимптотические улучшения) были настолько длинными и сложными, что для их создания и проверки потребовались бы тысячи лет, а для их хранения понадобилось бы больше битов памяти, чем атомов во Вселенной”. Следовательно, недавние усилия по реализации систем ZK для общих вычислений посвящены альтернативным методам, которые не реализуют всех перечисленных выше свойств, хотя некоторые из них чрезвычайно эффективны на практике для конкретных размеров схем и для амортизированных вычислений.