**Vocabulary Study**

|  |  |  |
| --- | --- | --- |
| **Key word** | **Context** | **Translation** |
| *Subject* |  |  |
| *Theme* |  |  |
| *Topic* |  |  |
| *Issue* | Fortunately, we observe that the syntactic mismatch of the messages sent by and those sent by is the only **issue** preventing us from concluding that MIP projection and PCP evaluation are inverses of each other. [3:31] | К счастью, синтаксическое несоответствие сообщений, посылаемых и , является единственной **проблемой**, не позволяющей сделать вывод о том, что MIP-проекция и PCP-оценка являются инверсиями друг друга. |
| *Theory* | Barrington’s theorem is a fundamental result in computational complexity **theory** that has important implications for circuit design. [2:17] | Теорема Баррингтона — фундаментальный результат **теории** сложности вычислений, имеющий важные последствия для схемотехники. |
| *Model* | Miden VM and Miden rollup aim to achieve both goals by having the ability to have private state with private (local) execution in addition to the classical shared public state **model**. [1:1] | Miden VM и Miden rollup направлены на достижение обеих целей за счет возможности иметь частное состояние с частным (локальным) исполнением в дополнение к классической **модели** общего публичного состояния. |
| *Principle* |  |  |
| *Aspect* | That said, parallel repetition of 2-prover MIPs (with non-trivial soundness error) does decrease soundness error exponentially fast, at a rate that depends on certain **aspects** of the MIP [Raz95]; the rate of decay is studied and optimized in a line of works [Hol07; Rao11; BRRRS09; RR12]. [3:8] | Тем не менее, параллельное повторение MIP с двумя проверяющими (с нетривиальной ошибкой состоятельности) действительно уменьшает ошибку состоятельности экспоненциально быстро, со скоростью, зависящей от некоторых **аспектов** MIP [Raz95]; скорость уменьшения изучается и оптимизируется в ряде работ [Hol07; Rao11; BRRRS09; RR12]. |
| *Property* | The next lemma from [16] ensures that this functions satisfy the almost homomorphic **property** we referred to before. [4:20] | Следующая лемма из [16] гарантирует, что эти функции удовлетворяют **свойству** почти гомоморфности, о котором мы говорили ранее. |
| *Feature* |  |  |
| *Characteristics* |  |  |
| *Characteristic* | This **characteristic** causes the MCC to be more resource-intensive than other protocols. [5:2] | Эта **особенность** обусловливает более высокую ресурсоемкость MCC по сравнению с другими протоколами. |
| *Characterize* | We comment on the proof of Theorem 2, which **characterizes** the limiting behavior of parallel repetition of a PCP in terms of the soundness error of its MIP projection. [2:11] | Прокомментируем доказательство теоремы 2, которая **характеризует** предельное поведение параллельного повторения PCP в терминах ошибки состоятельности его MIP-проекции. |
| *Peculiarity* |  |  |
| *Peculiar* |  |  |
| *Specific* | We discuss how parallel repetition behaves for a **specific** class of problems, constraint satisfaction problems (CSPs). [3:34] | Мы рассмотрим, как ведет себя параллельное повторение для **определенного** класса задач — задач удовлетворения ограничений (CSP). |
| *Specifically* | **Specifically**, observing that all these constructions follow a common pattern, which we refer to as the sorting paradigm, we identify the key subprotocol (sorting check) that differentiates these constructions. [5:3] | **В частности**, наблюдая, что все эти конструкции подчиняются общему шаблону, который мы называем парадигмой сортировки, мы выделяем ключевой подпротокол (проверка сортировки), который отличает эти конструкции. |
| *Unique* | Every element in M, R, C has a **unique** representation. [2:8] | Каждый элемент в M, R, C имеет **единственное** представление. |
| *Common* | Specifically, observing that all these constructions follow a **common** pattern, which we refer to as the sorting paradigm, we identify the key subprotocol (sorting check) that differentiates these constructions. [5:3] | В частности, наблюдая, что все эти конструкции подчиняются **общему** шаблону, который мы называем парадигмой сортировки, мы выделяем ключевой подпротокол (проверка сортировки), который отличает эти конструкции. |
| *Special / specially* | One can easily check that homomorphic commitments are a **special** case of malleable commitments for which M, R and C are all groups. [2:8] | Легко проверить, что гомоморфные обязательства являются **частным** случаем податливых обязательств, для которых M, R и C — все группы. |
| *Particular particularly* | In **particular**, we know that for every unsatisfiable symmetric CSP , implies . [3:37] | В **частности**, известно, что для каждого неудовлетворительного симметричного CSP , подразумевает . |
| *Appropriate* | Nevertheless, an **appropriate** packed generalization of these schemes could potentially make them useful for our construction if the setting where and for security parameter , leading to negligible soundness error. [4:6] | Тем не менее, **соответствующее** пакетное обобщение этих схем может потенциально сделать их полезными для нашей конструкции в случае, когда и для параметра безопасности , что приведет к пренебрежимо малой погрешности. |
| *Scope* |  |  |
| *Pattern* | Specifically, observing that all these constructions follow a common **pattern**, which we refer to as the sorting paradigm, we identify the key subprotocol (sorting check) that differentiates these constructions. [5:3] | В частности, наблюдая, что все эти конструкции подчиняются общему **шаблону**, который мы называем парадигмой сортировки, мы выделяем ключевой подпротокол (проверка сортировки), который отличает эти конструкции. |
| *Sample* | The challenger **samples** some randomness , and then computes . [2:23] | Претендент **выбирает** некоторую случайность , а затем вычисляет . |
| *Way* | Indeed, suppose by **way** of contradiction that . [3:10] | Действительно, предположим в **порядке** противоречия, что . |
| *Technique* | Parallel repetition refers to a set of valuable **techniques** used to reduce soundness error of probabilistic proofs while saving on certain efficiency measures. [3:1] | Под параллельным повторением понимается набор ценных **приёмов**, используемых для уменьшения погрешности состоятельности вероятностных доказательств при сохранении определенных показателей эффективности. |
| *Approach* | However, this restriction is lifted by our third **approach** based on branching programs. [2:21] | Однако это ограничение снимается нашим третьим **подходом**, основанным на ветвящихся программах. |
| *Method* | There are efficient uniform sampling **methods** for any subgroups of M, R. [2:8] | Существуют эффективные **методы** равномерной выборки для любых подгрупп M, R. |
| *Methodology* |  |  |
| *Research* | It is unclear how to exploit the KZG-specific optimizations in grand-sum check, which is interesting for future **research**. [5:11] | Неясно, как использовать специфические для KZG оптимизации при проверке суммы, что представляет интерес для будущих **исследований**. |
| *Study (v/n)* | In this paper we initiate the **study** of parallel repetition for probabilistically checkable proofs (PCPs). [3:1] | В данной работе мы начинаем **изучение** параллельного повторения для вероятностно проверяемых доказательств (ВПД). |
| *Investigate* | This is a challenge that is worth **investigating** in future research. [2:20] | Эта проблема заслуживает **изучения** в будущих исследованиях. |
| *Investigation* |  |  |
| *Explore* | A variant of that technique making use of the malleability rather than the homomorphic property is an interesting question to **explore**. [2:20] | Вариант этой техники, использующий не гомоморфное свойство, а податливость, - интересный вопрос для **исследования**. |
| *Exploration* |  |  |
| *Analyse/analyze* |  |  |
| *Analysis* | The **analysis** of the counting problem is postponed to Appendix C. [3:39] | **Анализ** проблемы счета откладывается до Приложения C. |
| *Examine* |  |  |
| *Examination* | To address the above issues, it is crucial to formalize the problem of MCC and to conduct a systematic **examination** of existing solutions, which can help deepen our understanding of the problem, eliminate potential security risks, and identify and address the performance bottleneck. [5:2] | Для решения вышеуказанных вопросов крайне важно формализовать проблему MCC и провести систематическое **исследование** существующих решений, что поможет углубить наше понимание проблемы, устранить потенциальные риски безопасности, выявить и устранить узкое место в производительности. |
| *Consider* | For example, **consider** an addition gate with inputs , and output . [2:15] | Например, **рассмотрим** шибер сложения с входами , и выходом . |
| *Consideration* | Although existing lookup argument constructions [GW20, ZBK+22, PK22, GK22, ZGK+22, EFG22, Hab22, CBBZ22, SLST23] do not involve the selector vector , they can be adapted to take into **consideration**. [5:17] | Хотя существующие конструкции поискового аргумента [GW20, ZBK+22, PK22, GK22, ZGK+22, EFG22, Hab22, CBBZ22, SLST23] не предполагают использования селекторного вектора , они могут быть адаптированы для **учёта** . |
| *Considerable considerably* | Setting means that a soundness error of requires to either use a privacy threshold and challenges of size or (if we want smaller privacy threshold and challenges) to use repetition to amplify soundness, both options incurring still in **considerable** (although polynomial in ) communication overhead. [4:12] | Задание означает, что для получения ошибки в необходимо либо использовать порог секретности и вызовы размера , либо (если мы хотим получить меньшие порог секретности и вызовы) использовать повторение для усиления секретности, причем оба варианта влекут за собой **значительные** (хотя и полиномиальные по ) коммуникационные накладные расходы. |
| *Account* | This cost estimate applies to each layer of the proof and takes into **account** the size of the matrices used in the computation. [2:19] | Эта оценка стоимости относится к каждому слою доказательства и **учитывает** размер матриц, используемых в вычислениях. |
| *Perspective* | Theorem 5.2 gives an additional **perspective**: the soundness error does not tend to 0 because the MIP projection of the PCP has soundness error 1. [3:27] | Теорема 5.2 дает дополнительную **перспективу**: ошибка состоятельности не стремится к 0, поскольку MIP-проекция PCP имеет ошибку состоятельности 1. |
| *Prospect* |  |  |
| *Case* | In contrast, in this **case**, we show that the soundness error tends to as tends to infinity (the desired behavior). [3:39] | Напротив, в данном **случае** мы показываем, что ошибка состоятельности стремится к по мере того, как стремится к бесконечности (желаемое поведение). |
| *View* |  |  |
| *Significance / significant* | SNARKs (Succinct Non-Interactive Arguments of Knowledge) and STARKs (Scalable Transparent Arguments of Knowledge) are cryptographic protocols that allow for the verification of computations without revealing any sensitive information about the inputs or intermediate values and with **significantly** less effort than running the full computations directly. [1:1] | SNARKs (Succinct Non-Interactive Arguments of Knowledge) и STARKs (Scalable Transparent Arguments of Knowledge) - криптографические протоколы, позволяющие проверять вычисления без раскрытия какой-либо конфиденциальной информации о входных или промежуточных значениях и с **существенно** меньшими усилиями, чем при непосредственном выполнении полных вычислений. |
| *Insignificance / insignificant* |  |  |
| *Essential essentially* | Zero-knowledge proofs for NP statements are an **essential** tool for building various cryptographic primitives and have been extensively studied in recent years. [2:1] | Доказательства с нулевым знанием для утверждений NP являются **важным** инструментом для построения различных криптографических примитивов и в последние годы активно изучаются. |
| *Important* | Barrington’s theorem is a fundamental result in computational complexity theory that has **important** implications for circuit design. [2:17] | Теорема Баррингтона — фундаментальный результат теории сложности вычислений, имеющий **важные** последствия для схемотехники. |
| *Current* | The Miden VM **currently** has an instruction MSTREAM which uses a pointer, located on the stack, to fetch a double-word from memory and overwrite the top 8 stack field element with it. [1:9] | **В настоящее время** в ВМ Miden имеется инструкция MSTREAM, которая использует указатель, расположенный на стеке, для выборки из памяти двойного слова и перезаписи им верхнего 8-го элемента поля стека. |
| *Burning* |  |  |
| *Topical* |  |  |
| *Modern* |  |  |
| *Up-to-date* |  |  |
| *State-of-the-art* | When used in MCCs, our new lookup argument reduces the number of online polynomial oracles by 2 to 10 and the number of online queries by 1 to 9, compared to the **state-of-the-art** construction . [5:40] | При использовании в ЦУПах наш новый аргумент поиска уменьшает количество онлайновых полиномиальных оракулов на 2-10, а количество онлайновых запросов на 1-9, по сравнению с **современной** конструкцией . |
| *General / generally* | The rate of decay in the **general** case is a major open problem. [3:4] | Скорость распада в **общем** случае является основной открытой проблемой. [3:4] |
| *Identify* | We **identify** a criterion that characterizes when parallel repetition reduces soundness error of a PCP to zero (in the limit). [3:5] | **Определяется** критерий, характеризующий, когда параллельное повторение сводит к нулю (в пределе) ошибку состоятельности ПСП. |
| *Identification* |  |  |
| *Assess* |  |  |
| *Assessment* |  |  |
| *Estimate* | We present in Table 3 the **estimated** proof sizes and the costs of the prover and the verifier after instantiating the PIOP with the KZG polynomial commitment scheme. [5:39] | В табл. 3 приведены **оценки** объема доказательств и затрат проверяющего и проверяемого после инстанцирования PIOP с полиномиальной схемой обязательств KZG. |
| *Estimation* |  |  |
| *Discuss* | Note however, that this restriction does not affect the proof system for branching program, which relies on an affine relation, as **discussed** in Sec. 6.3. [2:20] | Заметим, однако, что это ограничение не затрагивает систему доказательства ветвящейся программы, которая опирается на аффинное отношение, как это **рассматривается** в разделе 6.3. |
| *Discussion* |  |  |
| *Illustrate* | Figure 4.1 **illustrates** this step. [1:7] | Рисунок 4.1 **иллюстрирует** этот шаг. |
| *Illustration* |  |  |
| *Demonstrate* | As a result, the prover must **demonstrate**, in the online phase, that the committed permutation is consistent with and . [5:31] | В результате на интерактивной фазе проверяющий должен **доказать**, что зафиксированная перестановка соответствует и . |
| *Demonstration* |  |  |
| *Show* | We **show** that given two AGAMCs we can derive a new one via a direct product. [2:10] | Мы **показываем**, что из двух AGAMC можно получить новую через прямое произведение. [2:10] |
| *Exhibit* | Moreover, we **exhibit** an “ordered” 3-coloring for . [3:22] | Более того, мы **демонстрируем** "упорядоченную" 3-раскраску для . |
| *Perform* | Within each layer , both the prover and verifier need to **perform** malleability operation. [2:19] | В каждом слое как проверяющему, так и проверяемому необходимо **выполнить** операций над нечистотой. |
| *Performance* | Our primary focus centers on constructing schemes within this model and providing **performance** metrics in terms of the number of group actions, group elements, and set elements. [2:2] | Основное внимание мы уделяем построению схем в рамках этой модели и получению показателей **эффективности** по количеству групповых действий, элементов группы и элементов множества. |
| *Carry out* |  |  |
| *Conduct* | To address the above issues, it is crucial to formalize the problem of MCC and to **conduct** a systematic examination of existing solutions, which can help deepen our understanding of the problem, eliminate potential security risks, and identify and address the performance bottleneck. [5:2] | Для решения вышеуказанных вопросов крайне важно формализовать проблему MCC и **провести** систематическое исследование существующих решений, что поможет углубить наше понимание проблемы, устранить потенциальные риски безопасности, выявить и устранить узкое место в производительности. |
| *Calculate* |  |  |
| *Calculation* |  |  |
| *Compute* | Moreover, in both [15] and our construction, the shares are **computed** efficiently using a matrix with entries in making them suitable for our -protocols. [4:4] | Более того, и в [15], и в нашей конструкции доли эффективно **вычисляются** с помощью матрицы с записями в , что делает их подходящими для наших Σ-протоколов. |
| *Computation* | SNARKs (Succinct Non-Interactive Arguments of Knowledge) and STARKs (Scalable Transparent Arguments of Knowledge) are cryptographic protocols that allow for the verification of **computations** without revealing any sensitive information about the inputs or intermediate values and with significantly less effort than running the full **computations** directly. [1:1] | SNARKs (Succinct Non-Interactive Arguments of Knowledge) и STARKs (Scalable Transparent Arguments of Knowledge) - криптографические протоколы, позволяющие проверять **вычисления** без раскрытия какой-либо конфиденциальной информации о входных или промежуточных значениях и с существенно меньшими усилиями, чем при непосредственном выполнении полных вычислений. |
| *Measure* | We **measure** their performance based on two factors: the number of times building blocks and are used, and the number of online polynomial oracles and evaluation queries outside of these building blocks. [5:39] | Мы **оцениваем** их производительность по двум факторам: количеству использования строительных блоков Perm и Lookup и количеству онлайновых полиномиальных оракулов и оценочных запросов, не относящихся к этим строительным блокам. |
| *Measurement* |  |  |
| *Weigh* |  |  |
| *Weight* |  |  |
| *Derive* | More precisely, we **derive** proof systems for arithmetic circuits, rank-1 constraint systems and branching programs. [2:3] | Более точно, мы **выводим** системы доказательств для арифметических схем, систем ограничений ранга 1 и ветвящихся программ. |
| *Derivation* |  |  |
| *Equal* | Depending on the application, and might be **equal**. [2:7] | В зависимости от задачи и могут быть **равны**. |
| *Equation* | Apart from the above vector **equations**, the verifiers can also check more complex relations such as. [5:6] | Кроме приведенных выше векторных **уравнений**, верификаторы могут проверять и более сложные соотношения, такие как. |
| *Assume* | For encrypting and decrypting, we can **assume** that the associated data, plaintext and ciphertext are already loaded into a memory region described by a memory pointer, and we can from now on focus on the interactions involving only the first 2 components (figure 4). [1:7] | Для шифрования и дешифрования можно **считать**, что соответствующие данные, открытый текст и шифртекст, уже загружены в область памяти, описываемую указателем памяти, и в дальнейшем мы можем сосредоточиться на взаимодействии только первых двух компонентов (рис. 4). |
| *Assumption* | Finally, based on a different **assumption**, called rough order **assumption**, [7] defines a sound argument that allows to prove the existence of x and allow for adversarially chosen g, but this is not a proof of knowledge, achieving only standard soundness. [4:16] | Наконец, на основе другого **предположения**, называемого **предположением** грубого порядка, в [7] определен обоснованный аргумент, позволяющий доказать существование x и допустить состязательно выбранное g, но это не является доказательством знания, достигая лишь стандартной обоснованности. |
| *Find* | This “failure” of parallel repetition is common: we **find** that it occurs for a wide class of natural PCPs for NP-complete languages. [3:1] | Такой "сбой" параллельного повторения встречается часто: мы **обнаружили**, что он имеет место для широкого класса естественных PCP для NP-полных языков. |
| *Findings* |  |  |
| *Support (v/n)* | Another example is given by the **supports** of codewords of weight 4 in an extended Hamming code, which yield a Steiner system, and leads to constructions of soundness-optimal challenge sets for with for any and , , and again with . [4:30] | Другой пример — **опоры** кодовых слов веса 4 в расширенном коде Хэмминга, которые дают систему Штейнера и приводят к построению оптимальных по звуку наборов вызовов для с при любом и , , и снова при . |
| *Challenge (v/n)* | However, the private execution model introduces some new **challenges** related to the state. [1:1] | Однако частная модель исполнения влечет за собой ряд новых **проблем**, связанных с состоянием. [1:1] |
| *Provide* | PLONK [GWC19] **provides** an example construction of the permutation argument. [5:16] | В работе PLONK [GWC19] **приведен** пример построения аргумента перестановки. |
| *Provision* |  |  |
| *Affect* | On the other hand, performing an MIP projection and a PCP evaluation are “inverses” of each other: MIP is essentially the same proof system as the MIP projection of PCP, up to a minor syntactic difference in the verifier alphabet that does not **affect** the soundness error. [3:13] | С другой стороны, выполнение MIP-проекции и PCP-оценки являются "инверсиями" друг друга: MIP - это, по сути, та же самая система доказательства, что и MIP-проекция PCP, за исключением небольшого синтаксического различия в алфавите верификатора, которое не **влияет** на ошибку состоятельности. |
| *Effect* |  |  |
| *Include* | Note that in case we can **include** additional “superfluous” secrets so that we get an appropriate . [4:15] | Заметим, что в случае мы можем **включить** дополнительные "лишние" секреты, чтобы получить соответствующее . |
| *Inclusion* |  |  |
| *Involve* | Consider the following language that **involves** systems of linear equations. [3:42] | Рассмотрим следующий язык , **включающий** системы линейных уравнений. |
| *Exclude* | The masking vector **excludes** the case from the range check. [5:23] | Маскирующий вектор **исключает** случай из проверки диапазона. |
| *Exclusion* |  |  |
| *Develop* | More recently, in concurrent work to this one (to appear in Asiacrypt23), [35] **developed** this connection between secret sharing and -protocols, providing a general construction of -protocols from verifiable secret sharing. [4:2] | Совсем недавно в работе, параллельной этой (появится в Asiacrypt23), [35] была **развита** эта связь между секретным обменом и -протоколами, дано общее построение -протоколов из проверяемого секретного обмена. |
| *Development* | Although there is a lack of literature discussing the recent **developments** in ZKVMs [zkS22, TV22, Pol22, Scr22, Mid22, Ris22, GPR21], these ZKVMs are the result of more than ten years of progress in the field of verifiable computations (VC) [GGP10]. [5:11] | Несмотря на недостаток литературы, в которой обсуждаются последние **разработки** ZKVM [zkS22, TV22, Pol22, Scr22, Mid22, Ris22, GPR21], эти ZKVM являются результатом более чем десятилетнего прогресса в области верифицируемых вычислений (ВВ) [GGP10]. |
| *Design (v/n)* | In this work, we **design** an encryption scheme that is optimized for SNARKs/STARKs. [1:2] | В данной работе мы **разрабатываем** схему шифрования, оптимизированную для SNARKs/STARKs. |
| *Classify* |  |  |
| *Classification* |  |  |
| *Establish* | Or perhaps a different analysis may **establish** an alternative expression for the rate of decay that is better for smaller values of . [3:7] | Или, возможно, в результате другого анализа можно **получить** альтернативное выражение для скорости затухания, которое будет лучше для меньших значений . |
| *Establishment* |  |  |
| *Present* | To better understand the protocols **presented** in this work, we provide an overview of the underlying intuitions. [5:5] | Для лучшего понимания протоколов, **представленных** в данной работе, мы приводим обзор основных интуиций. |
| *Presentation* |  |  |
| *Attempt (v/n)* | This work gives the first **attempt** at constructing a post-quantum generic proof system from isogeny assumptions (the group action DDH problem). [2:1] | В данной работе предпринята первая **попытка** построения постквантовой обобщенной системы доказательств из предположений изогении (проблема DDH группового действия). |
| *Explain* | We focus on the memory component, whose checking protocol is the most challenging to design for reasons that will be **explained** later. [5:6] | Мы сосредоточились на компоненте памяти, протокол проверки которой является наиболее сложным для проектирования по причинам, которые будут **объяснены** позже. |
| *Explanation* |  |  |
| *Emphasize / emphasize* | However, we want to **emphasize** that the limit approaching 1 does not mean that the soundness error of the parallel repetition is 1 for some . [3:28] | Однако мы хотим **подчеркнуть**, что приближение предела к 1 не означает, что ошибка состоятельности параллельного повторения равна 1 для некоторого . |
| *Emphasis* |  |  |
| *Thesis* |  |  |
| *Hypothesis* |  |  |
| *Describe* | We **describe** the obstacles and challenges towards obtaining succinct proofs in Sec. 6.4. [2:2] | В разделе 6.4 мы **описываем** препятствия и проблемы на пути получения кратких доказательств. |
| *Description* | The above **description** glossed over domain separation between the two absorption phases of associated data and plaintext. [1:9] | В приведенном выше **описании** не было замечено разделения областей между двумя фазами поглощения — ассоциированных данных и открытого текста. [1:9] |
| *Relative / relatively* | -protocols are a widely utilized, **relatively** simple and well understood type of zero-knowledge proofs. [4:1] | -протоколы являются широко используемым, **относительно** простым и хорошо понятным типом доказательств с нулевым знанием. |
| *Relevant* | This notion is **relevant** because it e.g. captures encryption functions from several cryptosystems with homomorphic properties (Joye-Libert, Paillier), where and are the plaintext and randomness spaces. [4:19] | Это понятие **актуально**, так как оно, например, охватывает функции шифрования из ряда криптосистем с гомоморфными свойствами (Joye-Libert, Paillier), где и — пространства открытого текста и случайности. |
| *Abstract* |  |  |
| *Concrete* | Their simplicity and **concrete** efficiency makes them widely used in various cryptographic applications and protocols, such as digital signatures, group signatures or anonymous credential systems, as well as secure multiparty computation protocols. [4:1] | Благодаря своей простоте и **конкретной** эффективности они широко используются в различных криптографических приложениях и протоколах, таких как цифровые подписи, групповые подписи или анонимные мандатные системы, а также в безопасных многосторонних вычислительных протоколах. |
| *Complex* | However, to decouple the memory from this potentially **complex** procedure, we consider to be an arbitrary value set by the machine executor. [5:7] | Однако, чтобы отвязать память от этой потенциально **сложной** процедуры, мы рассматриваем как произвольное значение, задаваемое исполнителем машины. |
| *Quantitative* |  |  |
| *Qualitative* |  |  |
| *Accurate* |  |  |
| *Inaccurate* |  |  |
| *Rough* |  |  |
| *Roughly* | However, this bound does not rule out that one can share a larger secret “at **roughly** the same price” (even in the threshold case). [4:12] | Однако эта граница не исключает, что можно поделиться большим секретом "**примерно** по той же цене" (даже в пороговом случае). |
| *Approximate* |  |  |
| *Approximation* | Parallel repetition underlies many results in hardness of **approximation**, which rely on custom-made PCP constructions in which one of the steps is to apply Raz’s Theorem on parallel repetition for MIPs [Raz95]. [3:4] | Параллельное повторение лежит в основе многих результатов по трудности **приближения**, которые опираются на специальные конструкции PCP, в которых одним из шагов является применение теоремы Раза о параллельном повторении для MIP [Raz95]. |
| *Approximately* | However, for 256-bit memory, the Triton VM check costs **approximately** 20 to 30 fewer online polynomial oracles and evaluation queries. [5:39] | Однако для 256-битной памяти проверка ВМ Triton обходится **примерно** на 20-30 меньшим числом онлайновых полиномиальных оракулов и оценочных запросов. |
| *Precise* | To be more **precise**, the modification to the alphabetical order serves as a random (in the ROM) permutation of the input, ensuring that revealing a path in the tree does not reveal the location of the leaf. [2:11] | **Точнее** говоря, модификация алфавитного порядка служит случайной (в ПЗУ) перестановкой входных данных, гарантирующей, что раскрытие пути в дереве не раскроет местоположение листа. |
| *Precisely* | More **precisely**, we define the encryption and decryption algorithms. [1:3] | **Точнее говоря**, мы определяем алгоритмы шифрования и дешифрования. |
| *Precision* |  |  |
| *Exactly* | In those cases where , we parallel repetition for the PCP **exactly** follows the rate of decay for the corresponding MIP projection, for which there may be known results that apply (since parallel repetition for MIPs has been studied in a rich line of work). [3:29] | В тех случаях, когда , параллельное повторение для PCP в **точности** повторяет скорость распада для соответствующей проекции MIP, для которой могут быть известны применимые результаты (поскольку параллельное повторение для MIP изучалось в богатом ряду работ). |
| *Apparent* |  |  |
| *Potential* | Hence, it would be worthwhile to explore its **potential** applications in lattice-based cryptography, where such properties are useful, see e.g. [5]. [4:6] | Поэтому целесообразно исследовать **возможности** его применения в криптографии на основе решеток, где такие свойства являются полезными, см. например [5]. |
| *Principal* |  |  |
| *Rigorous* |  |  |
| *Comparative / comparatively* |  |  |
| *Especially* |  |  |
| *Primary / primarily* | Our **primary** focus centers on constructing schemes within this model and providing performance metrics in terms of the number of group actions, group elements, and set elements. [2:2] | **Основное** внимание мы уделяем построению схем в рамках этой модели и получению показателей эффективности по количеству групповых действий, элементов группы и элементов множества. |
| *Mainly* | In encryption, there are **mainly** three stages. [1:7] | В шифровании **в основном** выделяют три этапа. |
| *Mostly* | However, there is an absence of literature discussing recent advances in MCC, as the constructions have **mostly** been developed in a haphazard manner and tightly connected to their engineering projects. [5:2] | Однако в литературе отсутствуют сведения о последних достижениях в области МКК, поскольку конструкции **в основном** разрабатывались бессистемно и были тесно связаны со своими инженерными проектами. |
| *Largely* | The recent rapid development of ZKVMs has been **largely** aided by the introduction of lookup arguments [GW20, ZBK+22, PK22, GK22, ZGK+22, EFG22, Hab22, CBBZ22, SLST23], which have significantly boosted the efficiency of 32- and 64-bit MCCs. [5:12] | В последнее время бурному развитию ZKVM **во многом** способствовало введение аргументов поиска [GW20, ZBK+22, PK22, GK22, ZGK+22, EFG22, Hab22, CBBZ22, SLST23], которые значительно повысили эффективность 32- и 64-разрядных ЦУУ. |
| *Direct / directly* | SNARKs (Succinct Non-Interactive Arguments of Knowledge) and STARKs (Scalable Transparent Arguments of Knowledge) are cryptographic protocols that allow for the verification of computations without revealing any sensitive information about the inputs or intermediate values and with significantly less effort than running the full computations **directly**. [1:1] | SNARKs (Succinct Non-Interactive Arguments of Knowledge) и STARKs (Scalable Transparent Arguments of Knowledge) - криптографические протоколы, позволяющие проверять вычисления без раскрытия какой-либо секретной информации о входных и промежуточных значениях и с существенно меньшими усилиями, чем при **непосредственном** выполнении полных вычислений. |
| *Indirect / indirectly* |  |  |
| *Frequent / frequently* | However, in constructing a PIOP, we **frequently** encounter situations where it would be convenient to design a building-block subprotocol for proving statements that involve polynomial oracles, e.g., “given the oracle access to that was previously sent from the prover, satisfies certain property”. [5:15-16] | Однако при построении PIOP мы **часто** сталкиваемся с ситуациями, когда было бы удобно разработать блочный подпротокол для доказательства утверждений, включающих полиномиальные оракулы, например, "учитывая доступ оракула к , который был ранее отправлен от проверяющего, удовлетворяет некоторому свойству". |
| *Eventually* |  |  |
| *Ultimate / ultimately* |  |  |
| *Merely* |  |  |
| *Hardly ever* |  |  |
| *Possible / possibly* | Observe that is a 3CNF formula that consists of all **possible** clauses using three variables. [3:37] | Заметим, что - это формула 3CNF, состоящая из всех **возможных** предложений с тремя переменными. |
| *Probable / probably* |  |  |
| *Basically* |  |  |
| *Put forward* |  |  |
| *Make up / made up* |  |  |
| *Point out* | We do **point out** that it may be worthwhile to investigate whether the techniques from Section 6.1 using Galois rings can be used to construct proofs of plaintext knowledge in that case. [4:19] | Мы **отмечаем**, что, возможно, имеет смысл исследовать, можно ли в этом случае использовать приемы из раздела 6.1 с использованием колец Галуа для построения доказательств знания открытого текста. |
| *Point up* |  |  |
| *Set out* |  |  |
| *Go into* |  |  |
| *Go through* |  |  |
| *Amount* | Indeed, if every set of size contains the same **amount** of sets from , this **amount** must be : we have already argued that can contain at most such sets, but we also know, by definition of , that there are sets in whose union has size less than r and hence has to be contained in some set of size . [4:29] | Действительно, если каждое множество размера содержит такое же **количество** множеств из C, то это **количество** должно быть : мы уже утверждали, что может содержать не более таких множеств, но мы также знаем, по определению , что в существует множество, объединение которых имеет размер меньше r и, следовательно, должно содержаться в некотором множестве размера . |
| *Number* | Let the **number** of parallel repetitions of the base schemes Figs. 1 and 2 be λ in Fig. 3. [2:15] | Пусть на рис. 3 **число** параллельных повторений базовых схем рис. 1 и 2 равно . |
| *Substantial / substantially* |  |  |
| *Deal with* | We can also extend our construction to **deal with** proving knowledge of preimage of group elements via ZK-ready functions, see Theorem 8. [4:4] | Мы также можем расширить нашу конструкцию для **решения** задачи доказательства знания прообраза элементов группы через ZK-готовые функции, см. теорему 8. |
| *Devoted to* |  |  |
| *Concern (v/n)* |  |  |
| *Concerning* |  |  |
| *Few / a few* | Note that below we only consider graphs with because all graphs with **fewer** than 6 edges are 3-colorable. [3:21] | Заметим, что ниже мы рассматриваем только графы с , поскольку все графы с числом ребер **менее** 6 являются 3-колорируемыми. |
| *Little / a little* |  |  |
| *Occur* | This “failure” of parallel repetition is common: we find that it **occurs** for a wide class of natural PCPs for NP-complete languages. [3:1] | Такой "сбой" параллельного повторения встречается часто: мы обнаружили, что он **имеет место** для широкого класса естественных PCP для NP-полных языков. |
| *Occurrence* |  |  |
| *Refer to* | The characterization relies on whether a corresponding MIP, which we **refer to** as the MIP projection of the PCP, has non-trivial soundness error. [3:24] | Характеристика зависит от того, имеет ли соответствующая MIP, которую мы **называем** MIP-проекцией PCP, нетривиальную ошибку состоятельности. |
| *Reference* |  |  |
| *Shed light on* |  |  |
| *In light of* |  |  |
| *Highlight* | However, it is essential to **highlight** that the KO-EGA model implies the existence of numerous cryptographic primitives with significant relevance in real-world applications [6,14,5,21]. [2:2] | Однако необходимо **подчеркнуть**, что модель KO-EGA предполагает существование множества криптографических примитивов, имеющих существенное значение для реальных приложений [6,14,5,21]. |
| *Crucial* | Note here it is **crucial** has coordinates in , so that can be computed with black box access to the group operation and inversion. [4:13] | Отметим, что здесь **очень важно**, чтобы имело координаты в , чтобы можно было вычислить с помощью "чёрного ящика" доступа к групповой операции и инверсии. |
| *Decisive* |  |  |
| *Sufficient* | Triton VM [TV22] mitigates this issue by its MCC with full address space, which is **sufficiently** large to cover the functionality of 256-bit memory. [5:12] | ВМ Triton [TV22] снимает эту проблему за счет MCC с полным адресным пространством, которое **достаточно** велико, чтобы покрыть функциональность 256-битной памяти. |
| *Preliminary / preliminarily* |  |  |
| *Comprehensive / comprehensively* | Our study provides a **comprehensive** overview of the various techniques used to build MCCs. [5:40] | В нашем исследовании представлен **полный** обзор различных методик, используемых для построения МЦК. |
| *Convenient* | Nonetheless, they have a significant drawback as circuits are **inconvenient** to program for, especially when branching and loops are involved. [5:12] | Тем не менее, они имеют существенный недостаток, поскольку схемы **неудобны** для программирования, особенно при наличии ветвлений и циклов. |
| *Contribute* | Overall, our work **contributes** an initial set of results on a basic question about probabilistic proofs, which we believe merits further study due to its fundamental nature. [3:4] | В целом наша работа **предоставляет** первый набор результатов по одному из основных вопросов вероятностных доказательств, который, на наш взгляд, заслуживает дальнейшего изучения в силу своей фундаментальности. |
| *Contribution* | Our main **contributions** are as follows. [5:3] | Наш основной **вклад** состоит в следующем. |
| *Evidence* |  |  |
| *Suggest* | Inspired by the techniques covered in this systemization, we **suggest** improvements to existing protocols in two aspects: a novel MCC protocol Permem that costs fewer building blocks, and a new lookup argument also with improved efficiency. [5:40] | Вдохновившись методами, рассмотренными в этой систематизации, мы **предлагаем** усовершенствование существующих протоколов в двух аспектах: новый MCC-протокол Permem, в котором используется меньшее количество строительных блоков, и новый аргумент поиска, также обладающий повышенной эффективностью. |
| *Propose* | We consider the framework **proposed** by Castagnos and Laguillaumie. [4:15] | Мы рассматриваем фреймворк, **предложенный** Castagnos и Laguillaumie. |
| *Offer* | However, our construction does **offer** several advantages. [2:20] | Однако наша конструкция **имеет** ряд преимуществ. |
| *A great deal of* |  |  |
| *A wide range of* |  |  |
| *To … (some) extent* |  |  |
| *In a variety of ways* |  |  |
| *Aim* | Parallel repetition is a class of ideas **aimed** at reducing soundness error without increasing key efficiency measures such as round complexity or query complexity. [3:4] | Параллельное повторение — это класс идей, **направленных** на уменьшение ошибки обоснованности без увеличения ключевых показателей эффективности, таких как сложность раунда или сложность запроса. |
| *Purpose* | For our **purposes** we want more flexibility and show that we can “decouple” both parameters. [4:13] | Для наших **целей** мы хотим получить большую гибкость и показываем, что можем "развязать" оба параметра. |
| *Goal* | A central **goal** in this area is constructing probabilistic proofs with small soundness error (the maximum probability that any prover convinces the verifier to accept an instance that is not in the language). [3:4] | Одной из центральных **задач** в этой области является построение вероятностных доказательств с малой ошибкой состоятельности (максимальной вероятностью того, что любой проверяющий убедит проверяемого принять экземпляр, который не находится в языке). |
| *Objective* |  |  |
| *Intend* |  |  |
| *In terms of* |  |  |
| *With respect to* | When the ring of the circuit can be embedded as a subgroup , the proofs for a variety of gates can be significantly improved, both **with respect to** proof size and efficiency. [2:15] | Когда кольцо схемы может быть вложено в виде подгруппы , доказательства для различных ворот могут быть значительно улучшены как **в отношении** размера доказательства, так и **в отношении** эффективности. |
| *With regard to* |  |  |
| *In respect of* |  |  |
| *Outline* | We introduce constructions derived from a secret sharing scheme **outlined** in [15] and make improvements to reduce its share size. [4:12] | Мы представляем конструкции, полученные из схемы разделения секрета, **изложенной** в [15], и вносим улучшения для уменьшения размера ее доли. |
| *Summary* | In **summary**, we have. [4:11] | В **итоге** мы имеем. [4:11] |
| *Summarize* | Triton VM overcomes this issue by designing a new technique for showing that is sorted by any total order, which we **summarize** in Algorithm 4. [5:26] | Triton VM преодолевает эту проблему, разрабатывая новую технику для доказательства того, что отсортирован любым суммарным порядком, которую мы **обобщаем** в алгоритме 4. |
| *Obtain* | Let be the CSP **obtained** from . [3:37] | Пусть — CSP, **полученный** из . |
| *Determine* |  |  |
| *Produce* |  |  |
| *Production* |  |  |
| *Create* | Intuitively, we say that a commitment scheme is malleable if given a commitment for an unknown value m, anyone can **create** a second commitment for some related to . [2:7] | Интуитивно мы говорим, что схема обязательств является податливой, если, имея обязательство на неизвестное значение m, любой может **создать** второе обязательство на некоторое , связанное с . |
| *Creation* |  |  |
| *Make use of* |  |  |
| *Extensive / extensively* | Zero-knowledge proofs for NP statements are an essential tool for building various cryptographic primitives and have been **extensively** studied in recent years. [2:1] | Доказательства с нулевым знанием для утверждений NP являются важным инструментом для построения различных криптографических примитивов и в последние годы **активно** изучаются. |
| *Careful / carefully* |  |  |
| *Detailed* |  |  |
| *In detail* | Parallel repetition has been defined and studied for IPs and MIPs, as we review **in more detail** in Section 1.2. [3:4] | Параллельное повторение было определено и исследовано для ИС и МИП, что **более подробно** рассматривается в разделе 1.2. |
| *Thorough* |  |  |
| *Thoroughly* | This type of designs are called Steiner systems and denoted (note the change of order) and have been studied **thoroughly**. [4:30] | Такого рода конструкции называются системами Штейнера и обозначаются (обратите внимание на изменение порядка), и они **достаточно полно** изучены. |
| *Remarkable* |  |  |
| *Striking* |  |  |
| *Consist in* | Then, Decisional Diffie-Hellman (DDH) problem **consists in** returning to guess . [2:24] | Тогда задача Decisional Diffie-Hellman (DDH) **состоит в том**, что возвращает для угадывания . |
| *Consist of* | A sigma protocol is a three-move proof system for a relation **consisting of** oracle-calling PPT algorithms (, ), where is deterministic. [2:5] | Сигма-протокол — это трехходовая система доказательств для отношения , **состоящая из** вызывающих оракул PPT-алгоритмов (, ), где — детерминированный. |
| *Contain* | Furthermore, it is uncertain whether they **contain** vulnerabilities due to their lack of formal security analysis. [5:2] | Кроме того, из-за отсутствия формального анализа безопасности неясно, **содержат** ли они уязвимости. |
| *Comprise* |  |  |
| *Yield* | However, in some cases the same knowledge error can be also attained by strictly smaller families of challenges, which consequently **yield** -protocols with the same soundness and smaller communication than using all possible sets of size t as challenges. [4:28] | Однако в некоторых случаях та же ошибка знания может быть достигнута и с помощью строго меньших семейств вызовов, которые, следовательно, **дают** -протоколы с той же состоятельностью и меньшей связностью, чем при использовании в качестве вызовов всех возможных множеств размера . |
| *Generate* | When we can no longer assume **generates** a linear secret sharing on and , since only is guaranteed to be a -module. [4:20] | При мы уже не можем предполагать, что **порождает** линейное разделение секрета на и , поскольку только гарантированно является -модулем. |
| *Allow* | In their work collects the sets of challenges that **allow** to extract a secret from the corresponding transcripts. [4:5] | В работе Γ собираются наборы вызовов, **позволяющие** извлечь секрет из соответствующих расшифровок. |
| *Enable* | The goal of zk-VMs is to **enable** trustless and privacy-preserving execution of virtual machines. [1:1] | Цель zk-VM — **обеспечить** выполнение виртуальных машин без доверия и с сохранением конфиденциальности. |
| *Ensure* |  |  |
| *Permit* |  |  |
| *Modify / modification* | If we need to **modify** the proof as follows. [4:21] | Если , то доказательство необходимо **модифицировать** следующим образом. |
| *Alter / alteration* |  |  |
| *Enhance / enhancement* |  |  |
| *Similar / similarly* | **Similarly**, by multiplying by a proper factor, one can obtain a generating set for any subgroup of G and sample uniformly at random from the subgroup. [2:5] | **Аналогично**, умножая на соответствующий коэффициент, можно получить порождающее множество для любой подгруппы G и произвести равномерную случайную выборку из этой подгруппы. |
| *Similarity* |  |  |
| *Verify / verification* | Another difference is that the general construction of their work uses Verifiable Secret Sharing (VSS) and the -protocol check is done through the share correctness **verification** algorithm. [4:5] | Еще одно отличие состоит в том, что в общей конструкции их работы используется Verifiable Secret Sharing (VSS), а проверка -протокола осуществляется с помощью алгоритма **проверки** корректности доли. |
| *While* | **While** in earlier works [11] was taken to be prime, subsequent works [21,13,6] have considered other cases, such as being a power of a prime or a product of primes. [4:15] | **Если** в ранних работах [11] принималось простым, то в последующих работах [21,13,6] рассматривались другие случаи, например, является мощностью простого числа или произведением простых чисел. |
| *Whereas* |  |  |
| *Since (=as)* |  |  |
| *For (=as)* |  |  |
| *Initially* |  |  |
| *Finally* | **Finally**, note that sigma protocols can be used to build non-interactive zero-knowledge proofs of knowledge. [2:6] | **Наконец**, отметим, что сигма-протоколы могут быть использованы для построения неинтерактивных доказательств знания с нулевым знанием. |
| *Following* | **Following** round 1 of Fig. 1, say seed generates , for each . [2:12] | **Следуя** раунду 1 рис. 1, скажем, что seed генерирует , для каждого . |
| *Subsequently* | **Subsequently**, we build different flavors of generic proofs for NP statements. [2:20] | **Далее** мы строим различные варианты общих доказательств для утверждений NP. |
| *Although* | Setting means that a soundness error of requires to either use a privacy threshold and challenges of size or (if we want smaller privacy threshold and challenges) to use repetition to amplify soundness, both options incurring still in considerable (**although** polynomial in ) communication overhead. [4:12] | Задание означает, что для получения ошибки в необходимо либо использовать порог секретности и вызовы размера , либо (если мы хотим меньший порог секретности и вызовы) использовать повторение для усиления секретности, причем оба варианта влекут за собой значительные (**хотя** и полиномиальные по ) коммуникационные накладные расходы. |
| *Despite* | Given that, an adversarial prover can convince the verifier, with probability exactly , of the fact that is in the language, **despite** it is not. [4:31] | Учитывая это, противник может с вероятностью ровно , убедить проверяющего в том, что есть в языке **несмотря на то**, что это не так. |
| *In spite of* |  |  |
| *Instead (of)* |  |  |
| *Nevertheless* |  |  |
| *On the contrary* |  |  |
| *In contrast / by contrast* | **In contrast**, the proof size is improved by a factor of and the number of actions is reduced by a factor of . [2:15] | **Напротив**, размер доказательства улучшается в раз, а количество действий уменьшается в раз. |
| *On the one hand, on the other hand* | **On the other hand**, if the PCP has soundness error less than 1, then its MIP projection may or may not have soundness error less than 1. [3:11] | **С другой стороны**, если PCP имеет погрешность в звуке меньше 1, то его MIP-проекция может иметь погрешность в согласованности меньше 1, а может и не иметь ее. |
| *Furthermore* | **Furthermore**, we construct a proof system for a matrix branching relation. [2:2] | **Кроме того**, построена система доказательств для матричного отношения ветвления. |
| *Moreover* | **Moreover**, consider a simple PCP for this language. [3:9] | **Более того**, рассмотрим простейший PCP для этого языка. |
| *Due to* | **Due to** the validity of both transcripts, we have . [2:26] | **В силу** достоверности обеих транскрипций имеем . |
| *In order to* | A final permutation call is performed **in order to** generate the authentication tag T. [1:8] | **Для** генерации аутентификационной метки T выполняется заключительный перестановочный вызов. |
| *Hence* | The distributions and are identical for all , **hence** the scheme is statistical HVZK. [2:26] | Распределения and одинаковы для всех u, **следовательно**, схема является статистической HVZK. |
| *Therefore* | **Therefore**, . [2:26] | **Поэтому** . |
| *Provided that* |  |  |
| *Unless* | Second, we achieve statistical, rather than computational, zero-knowledge, as well as simulation extractability, meaning that an adversary cannot simulate a new proof **unless** they know a witness, regardless of the number of simulated proofs they have seen. [2:20] | Во-вторых, мы достигаем статистического, а не вычислительного нулевого знания, а также извлекаемости симуляции, то есть противник не может смоделировать новое доказательство, **если только** он не знает свидетеля, независимо от количества просмотренных им симуляций доказательств. |

**References:**

1. Tomer Ashur and Al Kindi. (2023). Arithmetization Oriented Encryption. *Cryptology ePrint Archive*. URL: <https://eprint.iacr.org/2023/1668.pdf>.
2. Mingjie Chen and Yi-Fu Lai and Abel Laval and Laurane Marco and Christophe Petit. (2023). Malleable Commitments from Group Actions and Zero-Knowledge Proofs for Circuits based on Isogenies. *Cryptology ePrint Archive*. URL: <https://eprint.iacr.org/2023/1710.pdf>.
3. Alessandro Chiesa and Ziyi Guan and Burcu Yıldız. (2023). On Parallel Repetition of PCPs. *Cryptology ePrint Archive*. URL: <https://eprint.iacr.org/2023/1714.pdf>.
4. Claudia Bartoli and Ignacio Cascudo. (2023). On Sigma-Protocols and (packed) Black-Box Secret Sharing Schemes. *Cryptology ePrint Archive*. URL: <https://eprint.iacr.org/2023/1652.pdf>.
5. Yuncong Zhang and Shi-Feng Sun and Ren Zhang and Dawu Gu. (2023). Polynomial IOPs for Memory Consistency Checks in Zero-Knowledge Virtual Machines. *Cryptology ePrint Archive*. URL: <https://eprint.iacr.org/2023/1555.pdf>.