

Topological Interpretation of Elliptic Curves

Topological Interpretation of Finite Cyclic Groups

Avneet Singh
Interplanetary Company UG
sshmatrix@proton.me

ABSTRACT

Elliptic curves form the backbone of most cryptography today and are expected to feature in the post-quantum world through Zero-Knowledge algorithms. Most literature on Elliptic curves starts from the definition of a cubic symmetric polynomial and builds upon the group theoretic interpretation of finite fields. This may be sufficient to grasp the mere necessary details for implementing Elliptic curves in practise, but not necessarily ideal if one wants to understand why Elliptic curve geometry is unique and useful. To most readers, it can appear that Elliptic curves were drawn from a magic hat. In paper will illustrate that this is clearly not the case and there exists a very legitimate reason for how and why the humanity ended up using Elliptic curves for cryptography.

INTRODUCTION

This paper is intended as a standalone appendix to the main paper titled 'Intuitive Interpretation of Non-Interactive Zero-Knowledge Cryptography' [1]. The main paper deals with pretty much the entire field of cryptography starting from basic RSA to advanced Zero-Knowledge algorithms. In interest of conciseness, several important aspects of Elliptic curves couldn't be discussed in detail in the main paper; this document is an attempt to address those shortcomings for the overtly inclined reader. We'll also touch some peripheral topics that have tied number theory and elliptic curves at the hip. Having said that, we'll follow the same theme as the main paper and avoid jargon like plague.

COUNTING STONES

In order to truly arrive at present day cryptography in a natural sense, we'll start from the very basics of it all. Let's talk about numbers; numbers form the building blocks of cryptography and most mathematics, physics and natural sciences.

To naturally arrive at numbers and their geometry/topology, we will start from a simple yet abstract example of a collection of some stones circa 20,000 BC. Bob's tribe needs stones to make their flint tools and Bob is one of the stone gatherers; this is Bob's story. During this exercise, please refrain from using numbers as you know it in your subconscious. Try to think like Bob, who has no concept of 1, 2, 3 ... etc.

Consider a collection of stones that are of **approximately** the same size, shape, weight and texture such that Bob is unable to objectively distinguish any one stone from the rest despite each stone being unique. To quantify this collection of stones into **succinct** information¹ leads Bob into questioning the

¹so that Bob can convey information succinctly to others without having to show them the stones

nature of this collection. In order to **differentiate**/distinguish the stones among each other, Bob does the first natural thing – he names each stone. In other words, he assigns a unique label to each stone (see figure 1); such a system is a functioning naming system. Note that there is no condition on actual equivalence among any of the stones, and only an assertion that the observer is unable to tell them apart in a describable manner. Using this naming system (\aleph , 3, θ , k , τ , υ , A as labels), Bob can tell someone else what he saw by showing them the labels, assuming that the labels are an agreed upon standard that everyone recognises. Note that Bob can choose any 'ordering' of the labels and the communication system works the same. This works well for everyone until one day Bob finds a massive pile of stones, all of different shapes and sizes in large quantities. The problem is simply that Bob's naming standard has only a finite number of labels and the stones are too many to each correspond to one label. In other words, Bob has run out of labels. In this moment of tryst, Bob realises that he is not limited by the number of labels at all; he can put two labels together and make a new label! Now he just needs to prescribe a method of deciding which two numbers to put together, how to order them etc.

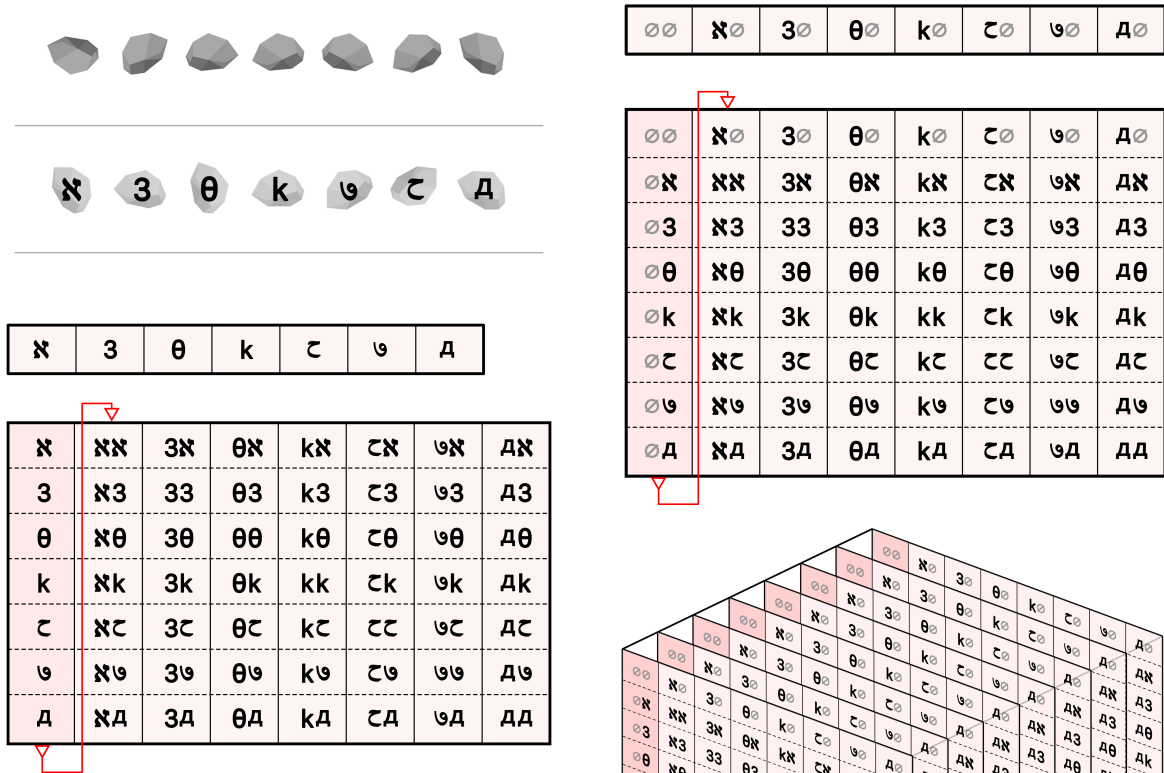


Figure 1: Bob's rudimentary number system

To follow Bob's footsteps, we start from \aleph until A , and then form the next label by joining: the first label with itself ($\aleph\aleph$), then first label with second label ($\aleph 3$), then first label with third label ($\aleph\theta$) and so on until Bob reaches ($\aleph A$). Bob proceeds to do this 'cycling' until he reaches (AA),

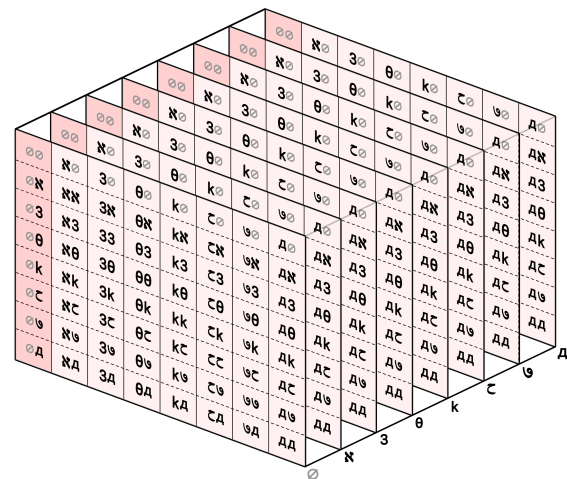


Figure 2: Symmetric number system after appending \emptyset label

at which point he can describe a lot more stones to his tribe than before. In figure 2, we can visualise Bob's naturally constructed number system. Readers

by now may have realised that this is exactly like our native **natural number** system with $(\aleph, 3, \theta, k, \tau, \vartheta, A) \rightarrow (1, 2, 3, 4, 5, 6, 7)$ with no presence of 8, 9 or 0, and it is nothing more. The terminology behind 'natural' is also a lot more intuitive to grasp. We can easily extend this system to count to AAA or $AAAA$ or $AAAAA$, extending up to any number of digits.

In computer science terminology, we know this as base- A (or base-7) encoding; base-10 is equivalent to the Decimal number system that we commonly use. We can however see that using a very small base quickly leads to overcrowding of digits required to describe any number of stones. On the other hand, a very high base requires keeping a lengthy shared standard among different parties in Bob's ecosystem.

DISCOVERING (+, -, ×, ÷)

In no time, Bob realises that his number system has some very remarkable properties. For instance, one day his friend Alice found $\aleph 3$ stones and Bob found θk stones. Typically he would start placing the combined set of stones on his lookup table (figure 1) and observing when he runs out of stones; then the label corresponding to the last stone is the total count of stones. This is an example of an **addition** operation that 'combines' or 'adds' or 'sums' the stones of Bob and Alice and counts them; let's represent this by $+$. Bob soon discovers though that he can be clever instead of following the rudimentary approach, i.e. he can describe the total quantity of stones by breaking the pairs $\aleph 3$ and θk apart into single digits, adding the first and second digits individually across both pairs and then re-attaching (denoted by $:$ operator) the individually summed digits in the same order. For example, $\aleph 3 + \theta k \rightarrow \aleph + \theta : 3 + k \rightarrow k : \vartheta \rightarrow k\vartheta$.

$$\aleph 3 + \theta k \rightarrow \aleph : 3 + \theta : k \rightarrow \aleph + \theta : 3 + k \rightarrow k : \vartheta \rightarrow k\vartheta$$

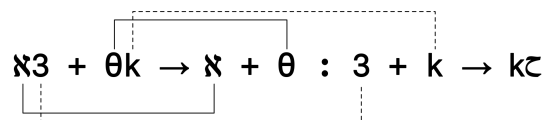


Figure 3: Bob's addition algorithm

In decimal representation, this is equivalent to $12 + 34 \rightarrow 1 : 2 + 3 : 4 \rightarrow 1 + 3 : 2 + 4 \rightarrow 4 : 6 \rightarrow 46$. Bob is ecstatic! But he soon finds that in some cases, the sum of the individual digits is made up of two digits, whereas his primitive algorithm expects a single digit output. For example, consider the addition $\aleph \tau + 3\vartheta \rightarrow \aleph + 3 : \tau + \vartheta \rightarrow \theta : \aleph k \rightarrow ?$. It appears that Bob is stuck, or is he? Bob makes a quick observation that he just needs to split the pair $\aleph k$ on the right again, keep the relevant second digit k and add the first digit \aleph to θ , such that

$$\aleph \tau + 3\vartheta \rightarrow \aleph + 3 : \tau + \vartheta \rightarrow \theta : \aleph k \rightarrow \theta + \aleph : k \rightarrow kk.$$

In decimal representation, this is equivalent to $15 + 26 \rightarrow 1 + 2 : 5 + 6 \rightarrow 3 : 11 \rightarrow 3 + 1 : 1 \rightarrow 4 : 1 \rightarrow 41$. What we have described here is the method of addition by 'carrying' which we have all been taught in primary schools when we

were kids, and it can be extended to add numbers with arbitrarily large digits. For example,

$$A\mathcal{Z} + 3\mathcal{U} \rightarrow A + 3 : \mathcal{Z} + \mathcal{U} \rightarrow \aleph\theta : \aleph k \rightarrow \aleph\theta + \aleph : k \rightarrow \aleph k : k \rightarrow \aleph k k, \text{ and}$$

In decimal representation, this is equivalent to $67 + 56 \rightarrow 6 + 5 : 7 + 6 \rightarrow 11 : 13 \rightarrow 11 + 1 : 3 \rightarrow 12 : 3 \rightarrow 123$. The end of algorithm is determined by the $:$ operator, i.e. when it has a single or pair to the left, at which point its function ends. In other words, the $:$ operator always starts with a pair to the right and ends with a single or pair to the left. For example,

$$\aleph A\mathcal{Z} + \theta 3\mathcal{U} \rightarrow k : \aleph\theta : \aleph k \rightarrow k : \aleph\theta + \aleph : k \rightarrow k + \aleph : k : k \rightarrow \mathcal{Z} k k.$$

Once Bob discovers addition, it is only a matter of time before he realises that it is a **commutative**/symmetric/order-agnostic operation, i.e. $\aleph + \theta \equiv \theta + \aleph$. The symmetric nature simply follows from the fact that the total count after adding Bob and Alice's stones is the same whether you start by putting Alice's stones first on the lookup table or Bob's. It is only natural from here on for Bob to discover **subtraction** ($\theta - \aleph \equiv 3 - 1 = 2$), which is simply the **inverse** of addition in the sense that it undoes the effects of addition. However, Bob notes that it is anti-commutative/asymmetric/order-dependent and it is in fact undefined when the order is reversed, i.e. $\aleph - \theta$. This is because Bob has no concept of negative integers (he is restricted to natural numbers) and it makes no intuitive sense for him to subtract/remove a larger count of stones from a smaller collection (e.g. $\aleph - \theta \equiv 1 - 3 = ?$).

With the ability to perform complex counting operations, Bob soon rises among the ranks of stone gatherers and starts his own business. Soon he employs hundreds of gatherers and becomes a wholesale stone supplier. With rising business however, he finds himself spending a lot of time doing lengthy operations, even after using his 'carry over' algorithm. One day he gets an unprecedented total of 345 stone gatherers, each collecting anywhere between 30 and 300 stones. This presented him with not only a huge number of evaluations/lookups, but also a lot of characters to write down. For instance, there were $\mathcal{Z}A$ individuals who brought $\aleph\theta A$ stones each, meaning Bob will need to make a record of it in the form of $(\mathcal{Z}A, \mathcal{Z}A, \dots \text{repeated } \aleph\theta A \text{ times } \dots, \mathcal{Z}A)$. To avoid this, Bob quickly makes an abbreviation $\mathcal{Z}A \times \aleph\theta A$ temporarily which he can expand later to evaluate in his own time. This is our native definition of **multiplication** which describes the repetition/replication count. While this notation is handy, it doesn't make the task of Bob easy right away. Bob still needs to expand his notation and then sum all the terms. In time, Bob realises that he now remembers some of the simpler evaluations by heart since he has done them thousands of times, e.g. $\mathcal{Z} \times A, \mathcal{U} \times k$ etc. Using his memory of simpler multiplications, Bob is able to perform his 'carry over' addition faster and soon doesn't need to use the expanded notation at all! He can simply denote:

$$(A\mathcal{Z} + A\mathcal{Z} + \dots \mathcal{U} \text{ repetitions } \dots) \equiv A\mathcal{Z} \times \mathcal{U} \rightarrow A:\mathcal{Z} \times \mathcal{U} \rightarrow A \times \mathcal{U} : \mathcal{Z} \times \mathcal{U},$$

followed by,

$$A \times \mathcal{U} : \mathcal{Z} \times \mathcal{U} \rightarrow \mathcal{U}A : \theta k \rightarrow \mathcal{Z}A + \theta : k \rightarrow \mathcal{U}\theta:k \rightarrow \mathcal{U}\theta k.$$

Using shorthand notation and lookup of simple multiplications in his memory, Bob is now able to calculate larger multiplications with arbitrarily large counts in the operation (stone count \times individuals with said stone count). In decimal notation, the above example is equivalent to:

$$(75 + 75 + \dots 6 \text{ repetitions } \dots) \equiv 75 \times 6 \rightarrow 7 : 5 \times 6 \rightarrow 7 \times 6 : 5 \times 6 \\ \rightarrow 42 : 30 .$$

x

FINDING ZERO THROUGH ARITHMETIC


Let's fast-forward a little bit and re-discover zero, i.e. 0. In figure 1, Bob's friend Arya notices that the horizontal and vertical axis are not the same 'stone units' in length, i.e. horizontal axis is one ($\rightarrow \aleph \rightarrow 1$) stone unit longer than vertical axis. Arya also notices that there is no formal label for describing complete absence of stones. He thereby suggests Bob that he should: **a)** add one more cell representing \emptyset ($\rightarrow \emptyset\emptyset \rightarrow \emptyset\emptyset\emptyset \dots$) at the top of the first column, and **b)** introduce labels of the form $\aleph\emptyset$, $3\emptyset$, $\emptyset\emptyset\dots A\emptyset$ such that the table's axis are equal in length. Bob however is not convinced since, **a)** he probably doesn't think that the 'absence of stones' needs a label, and secondly **b)** his intuitive interpretation of $\aleph\aleph$ following A (i.e. first stone in the first column of paired labels) doesn't work if $\aleph\emptyset$ happens to follow A . Bob resists the idea of \emptyset for the longest time; infact, the concept of zero originated much later than the basic arithmetic operations of addition (+), subtraction (-), multiplication (\times) and division (\div). This is not surprising since these operations do not need zero in their primal definitions. The presence of zero however makes complex combinations (aka 'functions') of (+, -, \times , \div) much easier to describe/evaluate despite being unintuitive. This eventually led to the acceptance of zero ($\rightarrow \emptyset \rightarrow 0$) at par with natural numbers about 18,000 years later. In essence, zero makes natural numbers complete or 'whole', thus leading to the term **whole numbers**. In our fictional present however, Bob finds zero too abstract and unintuitive, and doesn't include it in his number system. With time, Bob's number system finds utility in all aspects of life, and being able to count and do basic arithmetic using the 'stone table' lookup in figure 1 (which later became abacus) becomes equivalent to a Doctorate in Advanced Mathematics. Bob soon realises that he can not only count and describe objects using his system but he can quantify the 'size' of an object (once he assigns one stone unit to a specific size for setting a formal standard). By interpreting the table in 1 such that each cell now represents the 'size' of an object, Bob discovers the concept of length (units along a string or line), area (units on flat surfaces) and volume (units inside bulky objects). In his native number system, the definition of length is the same as counting stone units along a path, say for example A units. The definition of area (with side length A) in his system yields $(A + AA)$ units and the volume can be counted in $(A + AA + AAA)$ units.


REFERENCES


[1] Intuitive Interpretation of Non-Interactive Zero-Knowledge Cryptography


METADATA



Github: 

Contracts: 

Source: 

SHA-1 Checksum: 

Date: July 31, 2023