

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.

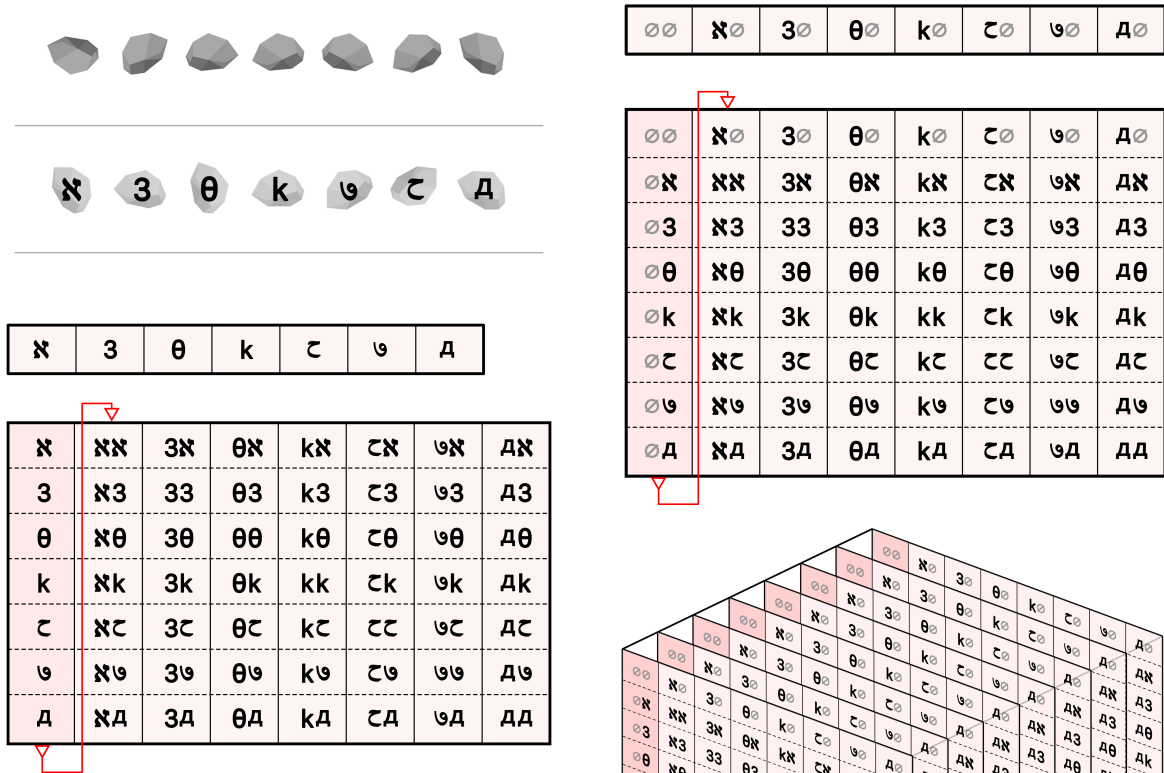


Figure 1: Bob's rudimentary number system

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),

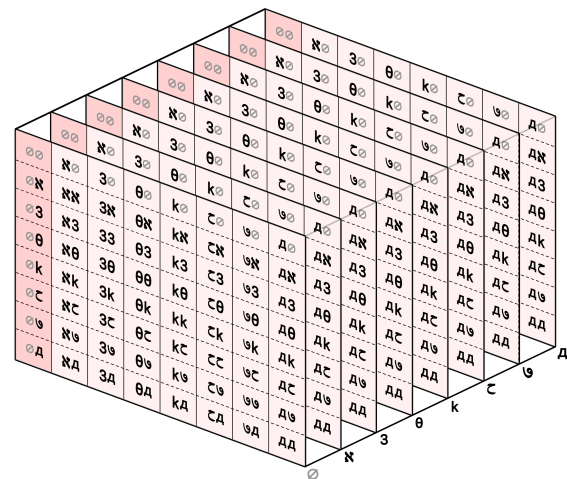


Figure 2: Symmetric number system after appending \emptyset label

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

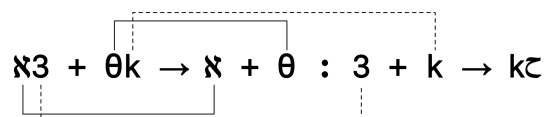
by now may have realised that this is exactly like our native **natural number** system with $(\aleph, 3, \theta, k, \tau, \vartheta, A) \rightarrow (1, 2, 3, 4, 5, 6, 7)$ with no presence of 8, 9 or 0, and it is nothing more. The terminology behind 'natural' is also a lot more intuitive to grasp. We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

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



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

Figure 3: Bob's addition algorithm

In decimal representation, this is similar 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 similar 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,

$$A\mathcal{C} + 3\mathcal{U} \rightarrow A + 3 : \mathcal{C} + \mathcal{U} \rightarrow \mathbb{N}\theta : \mathbb{N}k \rightarrow \mathbb{N}\theta + \mathbb{N} : k \rightarrow \mathbb{N}k : k \rightarrow \mathbb{N}kk, \text{ and}$$

In decimal representation, this is similar 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 and ends with a single or pair to the left. For example,

$$\mathbb{N}A\mathcal{C} + \theta 3\mathcal{U} \rightarrow k : \mathbb{N}\theta : \mathbb{N}k \rightarrow k : \mathbb{N}\theta + \mathbb{N} : k \rightarrow k + \mathbb{N} : k : k \rightarrow \mathcal{C}kk.$$

Once Bob discovers addition, it is only a matter of time before he realises that it is a **commutative**, symmetric and order-agnostic operation, i.e. $\mathbb{N} + \theta \equiv \theta + \mathbb{N}$. 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. This realisation quickly extended to **associativity** of addition, i.e. $\mathbb{N} + \theta + \mathcal{U} \equiv (\mathbb{N} + \theta) + \mathcal{U} = \mathbb{N} + (\theta + \mathcal{U})$. It is only natural from here on for Bob to discover **subtraction** ($\theta - \mathbb{N} \equiv 3 - 1 = 2$), 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, non-associative, asymmetric, order-dependent and it is in fact undefined when the order is reversed, i.e. $\mathbb{N} - \theta$. 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. $\mathbb{N} - \theta \equiv 1 - 3 = ?$).

With his newly found obsession of counting, Bob soon starts to remember most of the basic additions without explicitly needing to perform the stone table lookup. However, adding countably many stone collections remained a challenge, e.g. $(A\mathcal{C} + \mathcal{C}3 + A\mathcal{C} + A\theta + 3k + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + \dots) = ?$. While dealing with such lengthy evaluations, Bob notices that when there are several stone collections with the same count, his task is relatively quite easy e.g. $A\mathcal{C}$ featuring \mathcal{U} ($=6$) times in the above example. For instance, $(A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C})$ is easier to evaluate since Bob remembers what he gets when he adds A (and \mathcal{C}) repeatedly \mathcal{U} ($=6$) times; he has performed this operation thousands of times in the past. In addition, he starts writing long evaluations like $(A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C} + A\mathcal{C})$ as $A\mathcal{C} \times \mathcal{U}$, introducing the notation \times for describing repetitions in order to shorten his notation. 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 for \times at all! He can simply denote:

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

followed by,

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

In decimal notation, the above example is equivalent to:

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

$$\rightarrow 42 : 30 \rightarrow 42 + 3 : 0 \rightarrow 45 : 0 \rightarrow 450.$$

Curious bob immediately tests the multiplication operation for commutativity and associativity and finds that both properties hold; this doesn't come as a surprise to bob since he has constructed the \times operation from series of $+$ operations which are both atomically commutative as well as associative. 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 and memorised tables. One day he gets an unprecedented total of $\tau A3$ ($=345$ in decimal) stone gatherers, each collecting anywhere between $\aleph 3$ ($=9$) and $\aleph A\aleph$ ($=99$) stones. This presented him with a huge number of table lookups and evaluations. For instance, he noted that there were τA ($=60$) people who brought $\aleph \theta A$ ($=93$) stones each. Without his ' \times ' notation, Bob would need to make a record of it in the form of $(\tau A + \tau A + \dots \aleph \theta A \text{ times } \dots)$. However, Bob uses his clever trick and quickly makes the abbreviation $\tau A \times \aleph \theta A$ ($=60 \times 93$).

While the notation saves the day somewhat, it doesn't make Bob's job much easier. Unlike the previous (easier) example of 75×6 where Bob remembered the multiplication table of 6, in this case he doesn't know the multiplicative table of either τA ($=60$) or $\aleph \theta A$ ($=93$). While worried at the start, Bob soon realises that in fact only needs the tables of smallest atomic labels since $\tau A \times \aleph \theta A \rightarrow \tau \times \aleph \theta A : A \times \aleph \theta A$, where both the left and right side of $:$ can be individually evaluated as $\aleph \theta A \times \tau \equiv \aleph \times \tau : \theta \times \tau : A \times \tau$ and $\aleph \theta A \times A \equiv \aleph \times A : \theta \times A : A \times \tau$ respectively, such that

$$\begin{aligned} \tau A \times \aleph \theta A &\rightarrow \tau \times \aleph \theta A : A \times \aleph \theta A \rightarrow (\aleph \times \tau : \theta \times \tau : A \times \tau) : (\aleph \times A : \theta \times A : A \times A) \\ &\rightarrow (\tau : 3\aleph : kA) : (A : 3A : \theta A) \rightarrow (A : \tau : A) : (\aleph \theta : \theta : A) \rightarrow \\ &A\tau A : \aleph \theta \theta A \rightarrow A\tau A + \aleph \theta \theta : A \rightarrow \aleph 33\theta : A \rightarrow \aleph 33\theta A \end{aligned}$$

In decimal world, this is similar to

$$\begin{aligned} 62 \times 99 &\rightarrow 6 \times 99 : 2 \times 99 \rightarrow (6 \times 9 : 6 \times 9) : (2 \times 9 : 2 \times 9) \rightarrow (54 : 54) \\ &: (18 : 18) \rightarrow (54 + 4 : 4) : (18 + 1 : 8) \rightarrow (59 : 4) : (19 : 8) \rightarrow 594 : 198 \\ &\rightarrow 594 + 19 : 8 \rightarrow 6138. \end{aligned}$$

Using shorthand notation ' \times ' and lookup of simple multiplications in his memory, Bob is now able to calculate arbitrarily large multiplications with arbitrarily large arguments (stone count \times individuals with said stone count). What we have (re)constructed here is the well-known Vedic or Chinese or Japanese multiplication; one can imagine why there are so many names for it since it arises very naturally in any base system and was likely discovered independently by each intelligent civilisation.

Considering that Bob discovered the inverse of addition, it is natural for him to wonder if an inverse operation exists for multiplication; he denotes this operation with \div . In a countable sense, Bob interprets that **division** is equivalent to distributing a collection of stones equally among some individuals, e.g. he denotes his intention of dividing $A\theta$ stones among θ people with $A\theta \div \theta$. Bob initially deduces that division only makes sense if his collection of stones is sufficiently large to divide among any given group of people, i.e. for θ people, he needs at least θ stones to be able to give a

single stone to everyone. This situation is similar to subtraction where Bob wasn't able to subtract a bigger count from a smaller count; now he is unable to divide a smaller count by a bigger count. He finds that similar to subtraction, division is anti-commutative, non-associative, asymmetric and order-dependent. Bob lastly defines a corollary of division, the **modulo** operation $\%$, which describes how many stones will remain after Bob has finished equally distributing as many as he possibly can among all group members, e.g. $101 \% 6 = 5$.

PRIME COLLECTIONS

Bob's good days are yet again shortlived. One day, he is assigned the task of dividing the day's collection of 101 stones equally among 6 people in a group. Bob quickly finds that he is unable to do so; in decimal terms he had been asked to divide 101 stones among 6 people equally. He further notices that for a given group of people, some collections cannot be equally distributed. He also notes that for any given collection of stones, he can distribute it equally to only certain groups. In other words, not all numbers are divisible by other smaller numbers! This property is somewhat more bizarre to interpret for Bob; subtraction never caused him trouble in the sense that he could always distribute any collection of stones among any group of people as long as he didn't need to be fair and equal. Division seems to create more constraints for Bob when the condition of equality is imposed among all members of the group. Some days later, Bob's friend Arya noted that he was unable to equally divide his collection of 101 among $3 (=9)$ people. Bob finds it odd that neither he nor his friend could equally distribute their equivalent collections of $101 (=101)$ stones to distinctly different groups of people (with 6 and 9 members). Bob investigates more and finds that the collection of 101 stones is in fact impossible to distribute equally among members of any countable group²; we know these numbers as **prime numbers**. Bob and Arya's investigations reveal that these numbers continue to exist no matter how far you extend the natural numbers, and their distribution is base-agnostic and aperiodic.

FINDING ZERO THROUGH ARITHMETIC

Let's fast-forward a little bit and re-discover zero, i.e. 0 . In figure 1, Arya notices that the horizontal and vertical axis are not the same 'stone units' in length, i.e. horizontal axis is one ($\rightarrow 1 \rightarrow 1$) 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 0 ($\rightarrow 00 \rightarrow 000 \dots$) at the top of the first column, and,

b) introduce labels of the form $10, 30, 00 \dots 10$ such that the table's axis are equal in length.

Bob however is not convinced since,

i) he probably doesn't think that the 'absence of stones' needs a label, and secondly,

ii) his intuitive interpretation of 10 following 1 (i.e. first stone in the first column of paired labels) doesn't work if 10 happens to follow 1 .

Bob resists the idea of 0 for the longest time; infact, the concept of zero

²with the exception of distributing a single stone per person

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.


Let's take a sanity break and talk in 'normal' mathematical language. Note that in order to count to higher and higher values (more digits), Bob needs keep extending the dimensions of his 'counting box'. The 'counting volume' required to count up to γ digits is then written as a series sum $(A + AA + AAA + \dots \text{upto } \gamma \dots + AA \dots \gamma \dots A)$. This is extremely dirty. Due to the asymmetric-by-1 nature of Bob's natural number system, many identities tend to have unnecessarily complex forms. In a whole number system however, the counting volume for γ digits is simply $(AA \dots \gamma \dots A)$; this is equivalent to 10^γ digits in the decimal system. This finally convinces Bob to accept \emptyset in his counting system, thereby formalising whole numbers as the default counting framework.


REFERENCES


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

METADATA

Github: 

Contracts: 

Source: 

SHA-1 Checksum: 

Date: August 1, 2023 