# Geometry of Arithmetic Expressions: I. Basic Concepts and Unsolved Problems (Draft)

A preprint

**Mingli Yuan**
Swarma Research
Beijing, 100083
mingli.yuan@gmail.com

January 8, 2025

## Abstract

TODO

***K*eywords** arithmetic expressions, hyperbolic geometry

## Contents

## 1 Introduction

Can arithmetic expressions form a geometric space? In this paper, we present several examples of arithmetic expression spaces and examine their properties.

TODO

## 2 Basic concepts

### 2.1 Arithmetic expression

In order to define arithmetic expressions involving real numbers $\mathbb{R}$ in a rigorous way, we need to use a sophisticated type theory. However, in order to keep things simple and maintain clarity, we will start by using only production rules, but with certain semantic restrictions. We will also begin with rational numbers $\mathbb{Q}$ to avoid the difficulties inside real numbers $\mathbb{R}$ .

**Definition 2.1.** *An arithmetic expression $a$ over $\mathbb{Q}$ is a structure given by the following production rules:*

$$
\begin{aligned}
a &\longleftarrow x \\
a &\longleftarrow (a+a) \\
a &\longleftarrow (a-a) \\
a &\longleftarrow (a\times a) \\
a &\longleftarrow (a\div a)
\end{aligned} \tag{1}
$$

*where $x \in \mathbb{Q}$, and we denote this as $a \in \mathbb{E}[\mathbb{Q}]$.*

During the production process, we can obtain both a string representation and a tree representation of arithmetic expression $a$, where the two representations are equivalent. For instance, the string representation of $a$ might be:

$$(((((1\times 2)\times 2)-1)\times(2+1))-6) \tag{2}$$

and the parsed syntax tree is depicted in Figure 1.

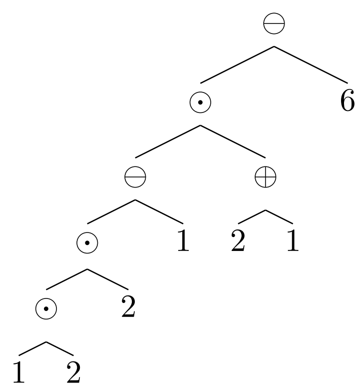


Figure 1: a tree representation of an arithmetic expression

If we interpret the target as a string and the building processes in production rule (1) as string building, we get the *string representation*. On the other hand, if the target is a tree, tree building leads to the *tree representation*. We can easily obtain the string representation of $a$ from its tree representation by performing a pre-order traversal.

The concept of a *sub-expression* can also be derived from the concept of a subtree. The branch nodes are all labeled with operators: $+$, $-$, $\times$, $\div$. The leaf nodes are all labeled with numbers.

Evaluation $\nu$ is a partial function that operates on arithmetic expression $a \in \mathbb{E}[\mathbb{Q}]$. It is undefined only if division by zero occurs during the recursive evaluation process.

We can define evaluation $\nu(a)$ of $a$ recursively as follows:

- Constant leaf: for any $x \in \mathbb{Q}$, $\nu(x) = x$.
- Compositional node by $+$: For any $(a+b)$, $\nu((a+b)) = \nu(a) + \nu(b)$.
- Compositional node by $-$: For any $(a-b)$, $\nu((a-b)) = \nu(a) - \nu(b)$.
- Compositional node by $\times$: For any $(a\times b)$, $\nu((a\times b)) = \nu(a)\nu(b)$.
- Compositional node by $\div$: For any $(a\div b)$, if $\nu(b) \neq 0$, then $\nu((a\div b)) = \nu(a)/\nu(b)$.

We say that an arithmetic expression $a$ is *evaluable* if $\nu(a)$ is defined. In the rest of this article, we will only consider evaluable arithmetic expressions unless stated otherwise.

Given an arithmetic expression $a$, whatever evaluable or not, we can obtain its tree representation. If a node $l$ is a leaf node, its corresponding subexpression $s$ is a number, so we consider it to be already "evaluated". If a node $b$ is a branch node, its corresponding subexpression $s$ is an expression, and we can apply $\nu$ to it to obtain a number $\nu(s)$. During the recursive evaluation process, starting from the leaves and moving towards the root, the subexpressions are evaluated one after another. However, the order of evaluations is generally not unique.

**Definition 2.2.** *The evaluation order of an arithmetic expression $a$ is an ordering of branch nodes in the tree representation of $a$ such that every node (sub-expression) is evaluated before its parent.*

For example, the possible evaluation orders of the arithmetic expression in Figure 1 are:

- $1 \times 2 \to \underline{2}; \underline{2} \times 2 \to \underline{4}; \underline{4} - 1 \to \underline{3}; 2 + 1 \to \underline{3}; \underline{3} \times \underline{3} \to \underline{9}; \underline{9} - 6 \to 3$
- $1 \times 2 \to \underline{2}; \underline{2} \times 2 \to \underline{4}; 2 + 1 \to \underline{3}; \underline{4} - 1 \to \underline{3}; \underline{3} \times \underline{3} \to \underline{9}; \underline{9} - 6 \to 3$
- $1 \times 2 \to \underline{2}; 2 + 1 \to \underline{3}; \underline{2} \times 2 \to \underline{4}; \underline{4} - 1 \to \underline{3}; \underline{3} \times \underline{3} \to \underline{9}; \underline{9} - 6 \to 3$
- $2 + 1 \to \underline{3}; 1 \times 2 \to \underline{2}; \underline{2} \times 2 \to \underline{4}; \underline{4} - 1 \to \underline{3}; \underline{3} \times \underline{3} \to \underline{9}; \underline{9} - 6 \to 3$

The underlined numbers are the numbers that are evaluated during the evaluation process.

Below are examples of expressions that have a unique evaluation order. These include right-expanded, left-expanded, and combinations of them, as shown in Figure 2 and Figure 3.

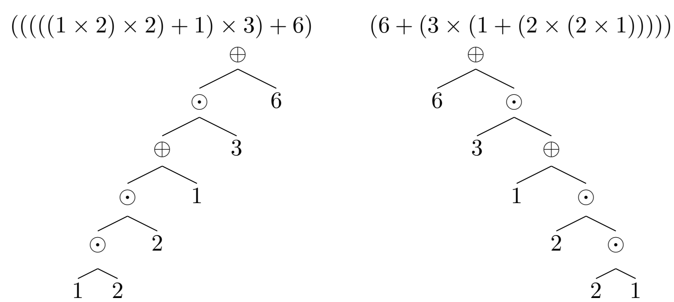


Figure 2: right-expanded and left-expanded expressions

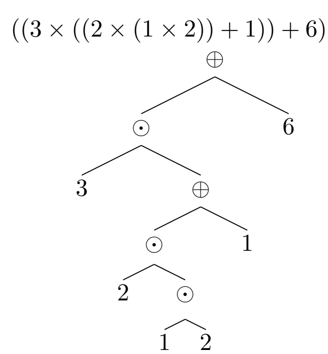


Figure 3: combinations of right-expanded and left-expanded expressions

The evaluation order of an arithmetic expression is related to the topological order of its tree representation, but they are not the same. The topological order of a tree is an ordering of nodes such that every node is visited before its parent[2]. However, we are only interested in the ordering of branch nodes, as leaf nodes have already been evaluated and can be ignored. Additionally, the topological order goes from parent to children, while the evaluation order goes from children to parent.

**Definition 2.3.** *A threadlike expression is an arithmetic expression that all the left nodes in its tree representation are leaf nodes.*

So a threadlike expression is right-expanded and its evaluation order is unique. One example of threadlike expressions is shown on the left side of Figure 2.

Threadlike expressions are significant here because they are analogous to the concept of paths in homotopy theory in geometry. In a more general context, certain special types of threadlike expressions are also interesting: for example, *alternating threadlike expressions* are expressions in which the additional and multiplicative operators appear in an alternating manner. In the field of computing, a hardware component called *multiplier-accumulator* (MAC) unit has been implemented [6], which is a special case of an alternating threadlike expression. As a result, some numerical algorithms based on MAC units have been studied [3].

### 2.2 A scalar field and a mesh grid

Consider the upper half plane $\{\mathcal{H} : (x, y)|y > 0\}$ equipped with an inner product and metrics defined as follows:

$$\mathbf{a} \cdot \mathbf{b} = \begin{bmatrix} a_x & a_y \end{bmatrix} \begin{bmatrix} \frac{1}{y^2} & 0 \\ 0 & \frac{1}{y^2 \ln^2 2} \end{bmatrix} \begin{bmatrix} b_x \\ b_y \end{bmatrix}$$

and

$$ds^2 = \frac{1}{y^2}(dx^2 + \frac{dy^2}{\ln^2 2})$$

We consider a scalar field satisfying

$$A = -\frac{x}{y} \tag{3}$$

We call this field an *assignment*.

Proper assignments allow us to establish a connection between paths in homotopy and threadlike arithmetic expressions, and to incorporate function theory into the study of arithmetic expression geometry.

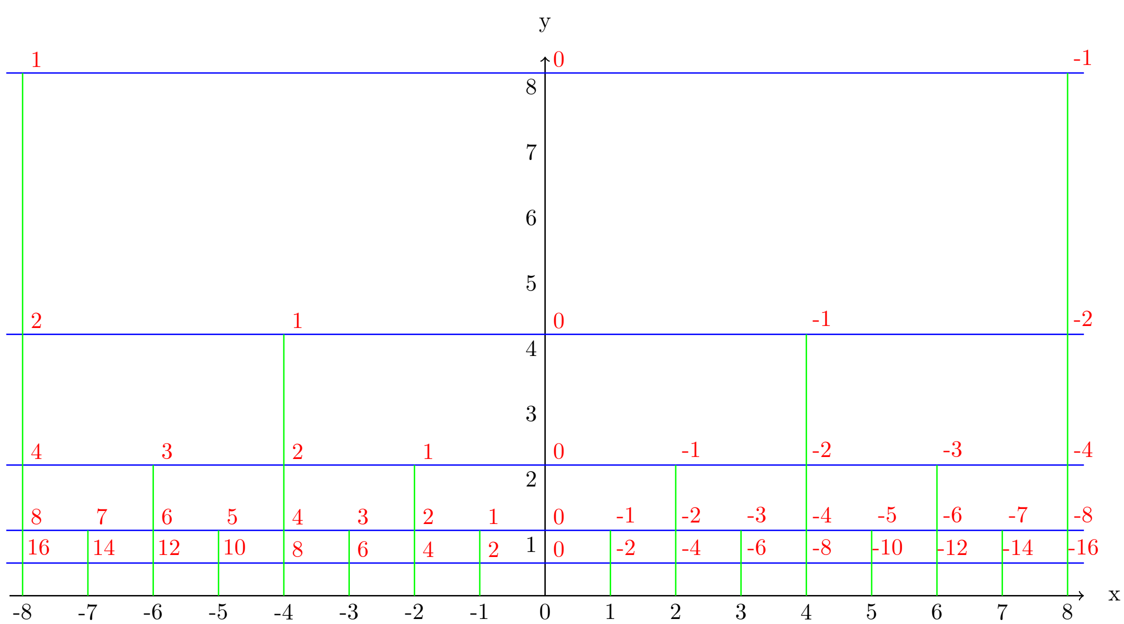


Figure 4: An addition-multiplication grid by generators with $\mu = 1$ and $\lambda = \ln 2$

We can draw a grid on the scalar field $A$ and underlying upper half plane $\mathcal{H}$ as shown in Figure 4. The blue lines encode a $+1$ relationship, the green lines encode a $\times 2$ relationship, and they are line families that are perpendicular to each other. The length of the line segments between two neighboring crossing points are unit length(calculations in lemma 8.1). The red value at the crossing points is the value of the scalar field $A$ at that point. Based on the relationships encoded by the lines, we can encode threadlike arithmetic expressions, which will be introduced in the subsection 2.3.

The addition-multiplication grid is also scale-invariant under the transformation

$$\begin{cases} x' = \alpha x \\ y' = \alpha y \end{cases}$$

where $\alpha = 2^k, k \in \mathbb{Z}$.

We can imagine if we make the grid finer and finer, the grid will become a continuous space. This leads to a rigorous treatment of arithmetic expressions as a geometric space in section 8.

### 2.3 Encoding threadlike expressions on the addition-multiplication grid

If we interpret the horizontal blue lines as $+1$ and the vertical green lines as $\times 2$ in Figure 4, we can encode threadlike expressions on the addition-multiplication grid. For example, in Figure 5 we encode $((((1 \times 4) - 1) \times 2) - 3)$ as the bold black lines.

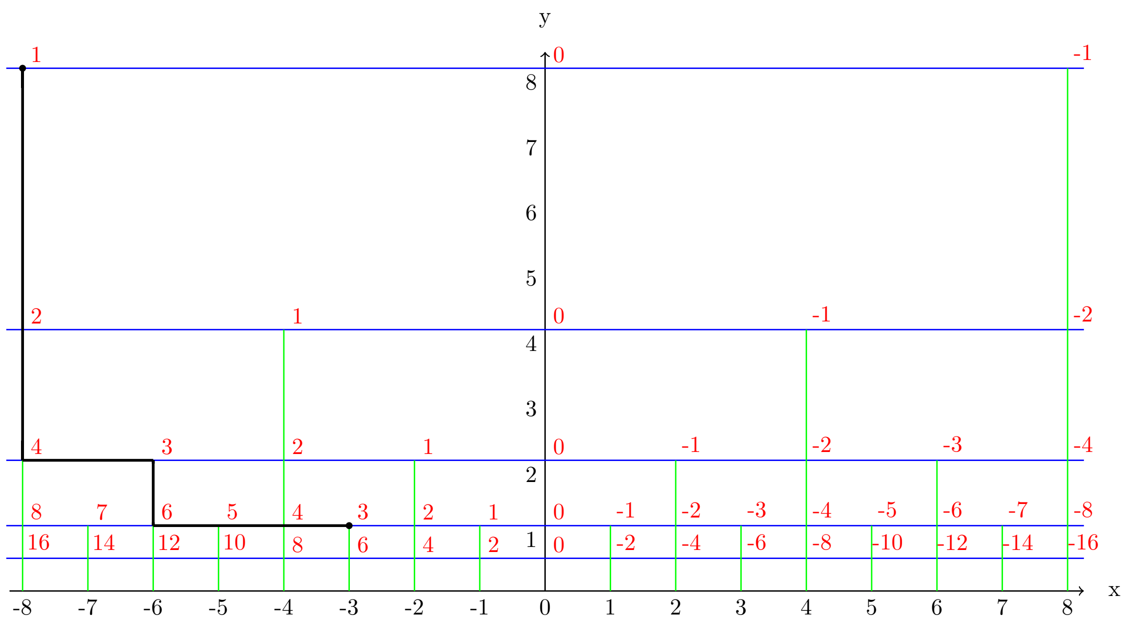


Figure 5: encoding threadlike expression

The zigzag lines in Figure 5 can be divided into four parts:

- the vertical line from 1 to 4: encoded as multiplication by 4
- the horizontal line from 4 to 3: encoded as subtraction by 1
- the vertical line from 3 to 6: encoded as multiplication by 2
- the horizontal line from 6 to 3: encoded as subtraction by 3

### 2.4 From a scalar field to a space of threadlike expressions

As shown in Figure 6, we have the following paths and expressions:

- the black path: $((1 \times 8) - 5) = 3$
- the purple path: $((1 - \frac{5}{8}) \times 8) = 3$

- the brown path: $((((((1-\frac{1}{8})\times 2)-\frac{1}{2})\times 2)-1)\times 2)=3$
- the orange path: infinite many addition-multiplication terms accumulated together, a special kind of integration

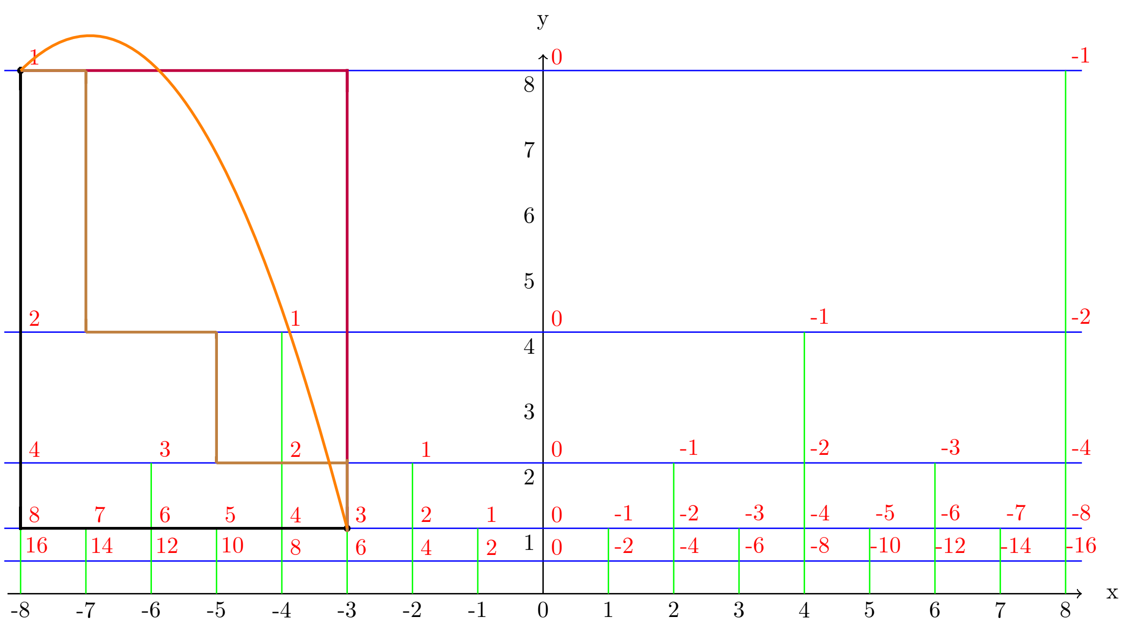


Figure 6: different encodings and their canonical form

All of the paths in Figure 6 have the same source 1 and same target 3. We will discuss a canonical form for these paths.

It is easy to see that the expressions can be transformed into each other by using the multiplication distributive law and by combining and decomposing terms.

Conversion form brown path to black path

$$3=((((((1-\frac{1}{8})\times 2)-\frac{1}{2})\times 2)-1)\times 2) \tag{4}$$

$$=1\times 8-\frac{1}{8}\times 8-\frac{1}{2}\times 4-1\times 2 \tag{5}$$

$$=((1\times 8)-5) \tag{6}$$

Conversion form brown path to purple path

$$3=((((((1-\frac{1}{8})\times 2)-\frac{1}{2})\times 2)-1)\times 2) \tag{7}$$

$$=(1-\frac{1}{8})\times 8-\frac{1}{2}\times 4-1\times 2 \tag{8}$$

$$=(1-\frac{1}{8})\times 8-\frac{1}{4}\times 8-\frac{1}{4}\times 8 \tag{9}$$

$$=(1-\frac{1}{8}-\frac{1}{4}-\frac{1}{4})\times 8 \tag{10}$$

$$=((1-\frac{5}{8})\times 8) \tag{11}$$

Therefore, we can define the black and purple paths in Figure 6 as a pair of canonical paths, which represent all threadlike expressions connecting the source 1 and the target 3.

Once we have such canonical paths, we can determine the canonical form of the whole space relative to an arbitrary source point $O$ and any other target point $P$. This allows us to define the space as a space of threadlike expressions.

### 2.5 Currying and path notation

Currying is a basic technique in functional programming[7], which is used to transform a function with multiple arguments into a sequence of functions with one argument. By currying a threadlike arithmetic expression, we can obtain a sequence of functions that operate on an operand, which is the leftmost leaf node.

We introduce the following notation for currying a threadlike arithmetic expression:

- initial operand: the leftmost leaf node
- operator: $\oplus_y : x \mapsto x + y$
- operator: $\ominus_y : x \mapsto x - y$
- operator: $\otimes_y : x \mapsto x \cdot e^y$
- operator: $\oslash_y : x \mapsto x \cdot e^{-y}$

For example, the threadlike arithmetic expression $(((((1 \times 2) \times 2) + 1) \times 3) + 6)$ can be curried as

$$\oplus_6(\otimes_{\ln 3}(\oplus_1(\otimes_{\ln 2}(\otimes_{\ln 2}(1)))))$$

Suppose we have a series of operators $a_1, a_2, \cdots a_{n-1}, a_n$, we introduce a *path notation*.

$$xa_1a_2 \cdots a_{n-1}a_n \coloneqq a\_n(a_{n-1}(\cdots a_2(a_1(x)) \cdots))$$

So, the above example can be written as

$$1 \otimes_{\ln 2} \otimes_{\ln 2} \oplus_1 \otimes_{\ln 3} \oplus_6$$

If a path begins with a number, we refer to it as a *bounded path*. If it does not, we refer to it as a *free path*, similar to the concept of vectors from the origin versus vectors at arbitrary points. a bounded path results in a number, while a free path results in a function.

Now we will verify that the operators within a path are associative.

**Lemma 2.1.** *The operators within a path are associative, i.e. we have*

$$a[bc] = [ab]c$$

*Proof.* We use normal typeface to express the path notation, and bold typeface to express the function notation.

For a free path, follow the definition, we have

$$a[bc] = [bc](\mathbf{a}) = \mathbf{c}(\mathbf{b}(\mathbf{a}))$$
$$[ab]c = \mathbf{c}([ab]) = \mathbf{c}(\mathbf{b}(\mathbf{a}))$$

hence, we have

$$a[bc] = [ab]c$$

is hold for a free path.

For a bounded path, we have

$$xa[bc] = [bc](\mathbf{a}(x)) = \mathbf{c}(\mathbf{b}(\mathbf{a}(x)))$$
$$x[ab]c = \mathbf{c}([ab](x)) = \mathbf{c}(\mathbf{b}(\mathbf{a}(x)))$$

hence, we have

$$a[bc] = [ab]c$$

is hold for a bounded path.

$\square$

**Definition 2.4.** *The concatenation of paths $p_1 \cdot p_2$ is defined as the composite of functions:*

$$p_1 \cdot p_2 \coloneqq p_2 \circ p_1$$

When a sequence of paths is concatenated, and only the first path can be bounded. If the first path is bounded, the concatenated result is a bounded path. Otherwise, the concatenated result is a free path.

### 2.6 Alternating threadlike expressions

Now we can define alternating threadlike expressions, which were mentioned in Section 2.1, using the path notion.

$$\alpha = a_1 b_1 a_2 b_2 \cdots a_l b_l, a_i = \otimes_{\lambda_i}, b_i = \oplus_{\mu_i}, \lambda_i, \mu_i \in \mathbb{R} \tag{12}$$

where $\bigoplus$ and $\bigotimes$ denote addition and multiplication, respectively, and the expression is a zigzag of alternating addition and multiplication operations. $\alpha$ is a free path, and we can bind a number to it.

Since 0 is the identity element for addition and 1 is the identity element for multiplication, it is straightforward to see that any arithmetic expression can be converted into an alternating threadlike expression by introducing more 0 and 1 into the original expression. So alternating threadlike expression is a kind of canonical form.

We can derive a formula for perturbations in alternating threadlike expressions.

Let us define the left-to-right accumulated sum of $\lambda_i$ as $\check{\lambda}_i$, such that:

$$\check{\lambda}_i = \sum_{j=1}^{i} \lambda_j, \check{\lambda}_0 = 0 \tag{13}$$

Then we also have right-to-left accumulated sum of $\lambda_i$

$$\hat{\lambda}_i = \check{\lambda}_l - \check{\lambda}_{l-i}, \hat{\lambda}_0 = 0 \tag{14}$$

Expanding equation (12) using the distributive law and the above notion at point $\mu_0$, we obtain:

$$\alpha(\mu_0) = e^{\lambda_l}(\cdots(e^{\lambda_2}(e^{\lambda_1}\mu_0 + \mu_1) + \mu_2)\cdots) + \mu_l \tag{15}$$

$$= e^{\hat{\lambda}_l}\mu_0 + e^{\hat{\lambda}_{l-1}}\mu_1 + e^{\hat{\lambda}_{l-2}}\mu_2 + \cdots + e^{\hat{\lambda}_1}\mu_{l-1} + e^{\hat{\lambda}_0}\mu_l \tag{16}$$

Next, at the starting point $\mu_0$, we introduce a perturbation $\tilde{\mu}_0 = e^{\eta_0}\mu_0 + \epsilon_0$, where $\eta_0$ and $\epsilon_0$ are the disturbance terms added by the summation and multiplication operations, respectively. Then, we have:

$$\alpha(\tilde{\mu}_0) = e^{\hat{\lambda}_l}(\tilde{\mu}_0) + e^{\hat{\lambda}_{l-1}}\mu_1 + e^{\hat{\lambda}_{l-2}}\mu_2 + \cdots + e^{\hat{\lambda}_1}\mu_{l-1} + e^{\hat{\lambda}_0}\mu_l \tag{17}$$

$$= \alpha(\mu_0) + e^{\hat{\lambda}_l}(\tilde{\mu}_0 - \mu_0) \tag{18}$$

As a result, purely from an arithmetic perspective, without the need for limits, we can derive the following meaningful ratio:

$$\frac{\alpha(\tilde{\mu}_0) - \alpha(\mu_0)}{\tilde{\mu}_0 - \mu_0} = e^{\hat{\lambda}_l} = e^{\check{\lambda}_l} \tag{19}$$

Now we extend this relationship from the starting point $\mu_0$ to the entire process, we define the recursive formula

$$w_i = e^{\lambda_i} w_{i-1} + \mu_i, w_0 = 0$$

and then we have

$$\frac{\tilde{w}_i - w_i}{\tilde{\mu}_0 - \mu_0} = e^{\check{\lambda}_i}, i \in \{1, ..., l\} \tag{20}$$

So, we have

$$\tilde{w}_i - w_i = e^{\check{\lambda}_i}(\tilde{\mu}_0 - \mu_0)$$

and hence

$$\tilde{w}_i - w_i = e^{\lambda_i}(\tilde{w}_{i-1} - w_{i-1}) \tag{21}$$

That means the perturbation along the path is controlled by the multiplication terms of $e^{\lambda_i}$.

### 2.7 Generated structure, commutator and arithmetic torsion

In order to study mesh grids like the one described in subsection 2.2, we need to investigate the algebraic structure of the threadlike arithmetic expressions that are generated.

For real number $\mathbb{R}$ and elements $\mu, \lambda \in \mathbb{R}$, we consider all the arithmetical expressions that are freely generated from

- initial operand: $0$
- operator: $\oplus_\mu : x \mapsto x + \mu$
- operator: $\ominus_\mu : x \mapsto x - \mu$
- operator: $\otimes_\lambda : x \mapsto x \cdot e^\lambda$
- operator: $\oslash_\lambda : x \mapsto x \cdot e^{-\lambda}$

We denote these expressions as $E(\mu, \lambda)$, where $\mu$ is the additional generator and $e^\lambda$ is the multiplicative generator. In cases where the context is clear, we may omit $\mu$ and $\lambda$ from the index. Our goal is not to study only a single $E(\mu, \lambda)$, but rather to use a family of $E(\mu, \lambda)$ to approach a continuous space.

Since $\oplus_\mu$ and $\ominus_\mu$ are mutually inverse operations, it follows that $\otimes_\lambda$ and $\oslash_\lambda$ are also mutually inverse. This means that $E(\mu, \lambda)$ forms a group. An observation is that the commutator of this group is not equal to identity generally, especially the commutator of the generators.

$$x \oplus_\mu \otimes_\lambda \ominus_\mu \oslash_\lambda - x = \mu(1 - e^{-\lambda}) \tag{22}$$

$$x \otimes_\lambda \oplus_\mu \oslash_\lambda \ominus_\mu - x = -\mu(1 - e^{-\lambda}) \tag{23}$$

Formula 22 obey the right-hand rule, and formula 23 obey the left-hand rule.

Or equivlently[1], we define below difference $\tau$ obey the right-hand rule:

$$\tau = x \oplus_\mu \otimes_\lambda - x \otimes_\lambda \oplus_\mu = \mu(e^\lambda - 1) \tag{24}$$

These differences are constant, indicating a type of torsion in the generated group. And torsion $\tau$ is specifically referred to as the arithmetic torsion.

We will reveal that $\tau$ is related to the curvature of the surface in later sections.

### 2.8 Problems on equality, singularity, symmetries

From the perspective of computer science, it is useful to consider different levels of equality within freely generated structures.

- Literal equality: the finest level of equality, judged by the string representation of the expression
- Syntactical equality: equality under certain syntactical rules
  - When inverse operators exist, it forms a group
  - When the commutative and distributive laws exist, it can be considered an algebra
- Semantic equality: the coarsest level of equality, judged by the evaluation of the expression

Literal equality is the strictest level of equality, and two different threadlike expressions are considered equal only if their string representations are exactly the same. This level of equality may be too strict, as it may not be compatible with the evaluation of the expression. However, under literal equality, the generated structure is the most rich and provides the base textures that can be woven into a space.

[1]Please reference section 3.4, the equivlence here is refered to the same order of the infinitesimal

Semantic equality is the least strict level of equality, and two different threadlike expressions are considered equal if they evaluate to the same number. This level of equality provides the total symmetrical resources of the space.

We can think of literal equality as the bottom and semantic equality as the top of a lattice, with syntactical equality being a compromise between the two extremes.

To end this introduction part of the paper, we present several problems and speculations that drives our research. These important problems arise from distance between syntactical and semantic structures.

*Foundational problem*: A careful reader may have noticed that the definition 2.1 is based on rational numbers $\mathbb{Q}$. Why can't we use real numbers $\mathbb{R}$ instead? The answer is that syntactically valid expressions may not be semantically valid. Dividing by zero can lead to invalid expressions, and the evaluation of the expression cannot be defined in this situation. Therefore, in real numbers, an expression may be syntactically valid but semantically not valid, and there is no algorithm that can decide whether an expression is semantically valid or not. How can we bridge this gap and provide a continuous geometry space? We will attempt to partially solve this problem in some special cases in section 8.

*Singular point problem*: We have a very strong intuition that semantically invalid expressions lead to singular points. The way we discussed in complex analysis may be borrowed here: essential singularities and poles.

*Symmetry and classification problem*: We conjecture that the equality lattice may not only play a role in the construction of a space, but also determine the symmetry of that space. We can imagine that, at certain levels of the lattice, we weave syntactically generated substructures into points to form a space, and the weaving process uses up some symmetrical resources, leaving the rest to form a symmetry on the space. The structure within the total symmetry may provide us with a systematic way of constructing spaces, and allow us to classify spaces based on their symmetries.

# 3 Flow equation

## 3.1 Flow equation

Consider an infinitesimal generating process on a Riemannian surface $M$ using two generators: one for an additional action $\mu$ and the other for a multiplicative action $e^{\lambda}$. These two generators are perpendicular. This generation process produces an assignment $A : M \to R$ over the surface.

For any point with an assignment $a_0$, if we consider a movement of distance $\epsilon$ in a direction with angle $\theta$ over a time period of $\delta$, we can establish the following:

$$a_\delta = (a_0 + \mu\epsilon\cos\theta)e^{\lambda\epsilon\sin\theta}$$

or

$$a_\delta = a_0 e^{\lambda\epsilon\sin\theta} + \mu\epsilon\cos\theta$$

Both formula can be simplified to the same result:

$$a_\delta = a_0 + \epsilon(a_0\lambda\sin\theta + \mu\cos\theta)$$

Then, we have the following equation:

$$\frac{1}{\delta}(a_\delta - a_0) = \frac{\epsilon}{\delta}(\mu\cos\theta + x_0\lambda\sin\theta)$$

When both $\delta$ and $\epsilon$ are towards zero, we get $da/dt$, and hence

$$\frac{da}{dt} = u(\mu\cos\theta + a\lambda\sin\theta)$$

Or, we can change it to another form

$$\frac{da}{ds} = \mu\cos\theta + a\lambda\sin\theta \tag{25}$$

We name this equation (25) as the flow equation.

The left side of this equation is governed by the distance structure, while the right side is governed by the angle structure. So that the isometrics of the surface keep the flow equation (25).

We can also get a direct formal solution of the flow equation (25)(details in Appendix A).

$$a = (a_0 + \frac{\mu}{\lambda}\cot\theta)e^{\lambda s\sin\theta} - \frac{\mu}{\lambda}\cot\theta \tag{26}$$

## 3.2 Discrete generating

In section 2.7, we have discussed a discrete generating process. Since flow equation governs an infinitesimal generating process, we will show the above discrete generating process can be emerged from the solution of the flow equation (26) naturally. We expand the formula by the Taylor series:

$$a = a_0 e^{\lambda s\sin\theta} + \frac{\mu}{\lambda}[1 + \lambda s\sin\theta + \frac{1}{2!}(\lambda s\sin\theta)^2 + \frac{1}{3!}(\lambda s\sin\theta)^3 + \cdots - 1]\cot\theta$$

Change the formula slightly:

$$a = a_0 e^{\lambda s\sin\theta} + \mu s\cos\theta + \frac{\mu}{\lambda}\sin\theta\cos\theta(\frac{\lambda^2 s^2}{2!} + \frac{\lambda^3 s^3}{3!}\sin\theta + \frac{\lambda^4 s^4}{4!}\sin^2\theta + \cdots)$$

By the formula of double angle, we have

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta + \frac{\mu}{2\lambda} \sin 2\theta (\frac{\lambda^2 s^2}{2!} + \frac{\lambda^3 s^3}{3!} \sin\theta + \frac{\lambda^4 s^4}{4!} \sin^2\theta + \cdots)$$

We denote

$$\Psi(s) = \frac{1}{2!} + \frac{\lambda s}{3!} \sin\theta + \frac{\lambda^2 s^2}{4!} \sin^2\theta + \cdots \tag{27}$$

Then we have

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta + \frac{\mu\lambda}{2} s^2 \Psi(s) \sin 2\theta \tag{28}$$

This formula gives the discrete generating process, when $\theta = \frac{k\pi}{2}, k = 0, 1, 2, 3 \cdots, s = 0, 1, 2, 3 \cdots$, we have

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta \tag{29}$$

Especially, we have the following four cases:

- $\theta = 0$: $a_s = a_0 + \mu s$
- $\theta = \frac{\pi}{2}$: $a_s = a_0 e^{\lambda s}$
- $\theta = \pi$: $a_s = a_0 - \mu s$
- $\theta = \frac{3\pi}{2}$: $a_s = a_0 e^{-\lambda s}$

This result is straightforward, but it demonstrates that the infinitesimal generating process is consistent with the discrete generating process. And this expands our toolset, enabling us to explore the interplay between discrete and infinitesimal generating processes.

### 3.3 The contour-gradient form of flow equation

It is easy to derive the contour equation in the local coordinate

$$\mu \cos\theta_c + a\lambda \sin\theta_c = 0 \tag{30}$$

then we have

$$\theta_c = -\arctan\frac{\mu}{a\lambda} \tag{31}$$

the contour and the gradient are perpendicular to each other

$$\theta_g = \pm\frac{\pi}{2} - \arctan\frac{\mu}{a\lambda} \tag{32}$$

then along $\theta_g$ we have

$$\frac{da}{ds} = \mu \cos(\pm\frac{\pi}{2} - \arctan\frac{\mu}{a\lambda}) + a\lambda \sin(\pm\frac{\pi}{2} - \arctan\frac{\mu}{a\lambda}) \tag{33}$$

$$\frac{da}{ds} = \pm\sqrt{\mu^2 + \lambda^2 a^2} \tag{34}$$

By introducing the right-hand rotation angle $\phi$ along the gradient direction, we can establish a local polar coordinate system based on the gradient and contour lines. Then the growth rate of $a$ along the angle $\phi$ is

$$\frac{da}{ds} = \mu \cos(\frac{\pi}{2} - \arctan\frac{\mu}{a\lambda} + \phi) + a\lambda \sin(\frac{\pi}{2} - \arctan\frac{\mu}{a\lambda} + \phi) \tag{35}$$

And the simplified equation is

$$\frac{da}{ds} = \sqrt{\mu^2 + a^2\lambda^2}\cos\phi \tag{36}$$

or

$$\frac{da_\phi}{ds_\phi} = \sqrt{\mu^2 + a^2\lambda^2}\cos\phi \tag{37}$$

if we want to emphasize the path is along the angle $\phi$.

The equation (36) is the flow equation in the contour-gradient coordinate system.

Equation (36) is solvable, and we get the relation between $a$ and $s$:

$$\tanh(\lambda s\cos\phi - c) = \frac{\lambda a}{\sqrt{\mu^2 + \lambda^2 a^2}} \tag{38}$$

we can further simplify the equation to

$$a = \pm\frac{\mu}{\lambda}\sinh(\lambda s\cos\phi - c) \tag{39}$$

Under the initial condition $a = a_0$ when $s = 0$, we can get the following equation:

$$a = \frac{\mu}{\lambda}\sinh(\lambda s\cos\phi + \operatorname{arcsinh}\frac{a_0\lambda}{\mu}) \tag{40}$$

or

$$a = -\frac{\mu}{\lambda}\sinh(\lambda s\cos\phi - \operatorname{arcsinh}\frac{a_0\lambda}{\mu}) \tag{41}$$

In this coordinate system, the additional line and the multiplicative line are:

$$\phi = \arccos\frac{\mu}{\sqrt{\mu^2 + a^2\lambda^2}} \tag{42}$$

$$\phi = \arcsin\frac{\mu}{\sqrt{\mu^2 + a^2\lambda^2}} \tag{43}$$

### 3.4 Arithmetic coordinate and area formula

We begin our exploration by examining the flow equation (25) within the framework of a local polar coordinate system:

$$\frac{da}{ds} = \mu\cos\theta + a\lambda\sin\theta \tag{44}$$

In an effort to re-contextualize this equation, we set $du = \cos\theta ds$ and $dv = \sin\theta ds$, where $du$ and $dv$ are perpendicular infinitesimal movements. We can use these movements to construct a local Descartes coordinate system, and the first fundamental form of this system is:

$$ds^2 = A^2du^2 + B^2dv^2 \tag{45}$$

Thereby this enables us to express the flow equation in a different light:

$$da = \mu du + a\lambda dv \tag{46}$$

Our attention now turns to the concept of arithmetic torsion, particularly at an infinitesimal level. Delving into the interplay between two infinitesimal generating processes, we observe that:

$$d\tau = (a_0 + \mu du)e^{\lambda dv} - (a_0 e^{\lambda dv} + \mu du) \tag{47}$$

From this relationship, we deduce:

$$d\tau = \mu du(e^{\lambda dv} - 1) \tag{48}$$

This leads us to an area formula, capturing the essence of this interaction:

$$d\tau = \mu\lambda du dv \tag{49}$$

and because the area element have a form

$$dS = |AB| du dv \tag{50}$$

Then we have

$$\frac{d\tau}{\mu\lambda} = \frac{dS}{|AB|} \tag{51}$$

This formula is compelling as it establishes a link between area elements and arithmetic torsion. Such formulations find parallels in the realms of classic analysis and differential geometry. For instance, they resonate with concepts akin to Stokes' theorem or the Gauss-Bonnet theorem. We intend to expand upon this formula in the ensuing section9, aiming to forge a connection with curvature and delve into the intricacies of the Gauss-Bonnet theorem.

It's noteworthy to emphasize the distinctiveness of the local Descartes coordinate system. This system, by integrating the assignment, lays the foundation for a theoretical framework. We refer to this as the *arithmetic coordinate system*, given its unique properties and alignment with arithmetic principles.

### 3.5 The coordinat-free form of flow equation

From the contour-gradient form of the flow equation (36), we can derive a coordinate-free form of the flow equation. Let's consider the direction of $\phi = 0$ in the contour-gradient coordinate system, and we have

$$\frac{da}{ds}|_{\phi=0} = \sqrt{\mu^2 + a^2\lambda^2}\cos 0$$

Notice the gradient of $a$ is not dependent on the coordinate system, and we have the coordinate-free form of the flow equation:

$$||\nabla a|| = \sqrt{\mu^2 + a^2\lambda^2} \tag{52}$$

It should be noted that the coordinate-free form of the flow equation (52) is an Eikonal equation, and can be viewed as a special Hamilton–Jacobi equation

$$H(x, a, \nabla a) = 0$$

where the Hamiltonian is

$$H(x, a, p) = ||p|| - \sqrt{\mu^2 + a^2\lambda^2} \tag{53}$$

### 3.6 Propagation method

Starting from Equations (40) and (41), we can derive a geometric propagation interpretation of the flow equation. By rewriting Equations (40) and (41) in a unified form, we obtain

$$a = \pm\frac{\mu}{\lambda}\sinh\big(\lambda s\cos\phi + \operatorname{arcsinh}\big(\tfrac{a_0\lambda}{\mu}\big)\big). \tag{54}$$

If we set $a_0 = 0$ and choose the gradient direction $\phi = 0$, the expression simplifies to

$$a = \pm\frac{\mu}{\lambda}\sinh(\lambda s). \tag{55}$$

Comparing Equation (55) with the circumference of a circle of radius $s$:

$$C = \frac{2\pi}{\sqrt{-k}}\sinh\big(\sqrt{-k}s\big), \tag{56}$$

we see that the assignment can be interpreted as propagating along a circle of radius. With forming a zero line, each point on this line generates a wavefront described by concentric circles. Hence, the propagation of corresponds to the wavefront evolution in this geometric sense. A more detailed and rigorous geometric interpretation of the flow equation will be provided in Section 4.

### 3.7 Flow and function

In this section, we aim to present novel insight into functions. Namely, the treatment of functions as flows will be discussed.

**Definition 3.1.** *Given a function $k$ on the real domain $R$, we can introduce a mapping $l$ on the arithmetic expression space $H$ such that the following diagram commutes.*

$$\begin{array}{ccc} H & \xrightarrow{l} & H \\ \nu\big\downarrow & & \big\downarrow\nu \\ R & \xrightarrow[k]{} & R \end{array}$$

*where $\nu$ is the evaluation function of the expression. Then we call the mapping $l$ is the promotion of the function $k$, or function $k$ is the projection of the mapping $l$.*

Giving an arithmetic expression space as definition at the beginning of the section2.2, we will show examples of flows as functions in the following Section 4.

### 3.8 The existence theorem

There are two existence problems related to the flow equation (25). The first existence problem concerns the existence of a function $a$ on a Riemannian surface $S$ that satisfies the flow equation (25). The second existence problem concerns the existence of a metric $g$ on a Riemannian surface $S$ that makes a function $a$ satisfy the flow equation (25).

We can proof there is a local morphing process over metric $g$ to make a function $a$ satisfy the flow equation (25) locally. But the global morphing process is more complicated, and we need to consider the global structure of the surface $S$, which is still not settled.

**Lemma 3.1.** *(By Le Zhang) Given an oriented compact Riemannian surface $S$, and a smooth function $a$ over $S$, there exists a metric $g$ on $S$ that makes $a$ satisfying the flow equation* (25)*.*

*Proof. Local perspective*:

Consider a point $p$ on the surface $S$, and there is a neighborhood $U$ around $p$. In this area, we can find a local isothermal coordinate system in which the metric takes the form:

$$ds^2 = e^{2\rho}(du^2 + dv^2),$$

where $u$ and $v$ are the coordinates of $U$, and $\rho$ is a function of $u, v$ in $U$. The gradient of $a$ in this local isothermal coordinate system is expressed as:

$$\nabla a = \frac{\partial a}{\partial u}du + \frac{\partial a}{\partial v}dv.$$

Using the definition of the directional derivative, we obtain:

$$\frac{da_\psi}{ds_\psi} = ||\nabla a||\cos\psi,$$

where $||\nabla a||$ is the norm of $\nabla a$, and $\psi$ is the angle between $\nabla a$ and the direction of movement.

Now, considering the flow equation 37 in the gradient-contour coordinate system, we have:

$$\frac{da_\phi}{ds_\phi} = \sqrt{\mu^2 + a^2\lambda^2}\cos\phi.$$

Note that $||\nabla a||$ is fixed for the given function $a$ and the local coordinate system, and $\sqrt{\mu^2 + a^2\lambda^2}$ is also fixed for the given function $a$. We can scale $e^{2\rho}$ with a linear factor $\alpha$ to make $||\nabla a||$ match the fixed value of $\sqrt{\mu^2 + a^2\lambda^2}$, thus we have a morphing process controlled by $\alpha$ that

$$ds^2 = \alpha e^{2\rho}(du^2 + dv^2) \tag{57}$$

$$= e^{2\rho + \ln\alpha}(du^2 + dv^2). \tag{58}$$

Under the morphing ratio $\alpha$, we have:

$$||\nabla_\alpha a|| = \alpha^{-1}||\nabla a||, \tag{59}$$

and when $\alpha$ is set to the value of:

$$||\nabla_\alpha a|| = \sqrt{\mu^2 + a^2\lambda^2},$$

the flow equation (25) is satisfied in the local coordinate system.

The morphing ratio $\alpha$ is calculated as follows:

$$\alpha = \frac{||\nabla a||}{\sqrt{\mu^2 + a^2\lambda^2}}. \tag{60}$$

□

When we consider the broader scope of the surface $S$, it's possible to extend the morphing process to every point, ensuring that the flow equation (25) is satisfied on a global scale. However, this expansion necessitates a harmonious integration of the morphing process across neighboring locales. Specifically, this means that the morphing should not only preserve the circles centered at point $p$ within its immediate local chart but also maintain the integrity of these circles within the adjacent charts of point $p$. In essence, the morphing process must be seamlessly coordinated across the various local regions to achieve a unified global transformation. How to achieve this harmonious integration remains an open question, and further exploration is needed to address this challenge.

# 4 The first kind arithmetic expression space $\mathfrak{E}_1$

In this section, we will present two analytic examples due to Le that are equivalent and belong to the class of spaces called the first kind arithmetic expression space $\mathfrak{E}_1$.

## 4.1 Example space I

Consider the upper half plane $\mathcal{H} : (x, y) \mid y > 0$ equipped with an inner product and metrics defined as follows:

$$\mathbf{a} \cdot \mathbf{b} = [a_x \quad a_y] \begin{bmatrix} \frac{1}{y^2} & 0 \\ 0 & \frac{1}{y^2} \end{bmatrix} [b_x \ b_y]$$

and

$$ds^2 = \frac{1}{y^2}(dx^2 + dy^2)$$

We also consider an assignment function $A$ defined on $\mathcal{H}$ as follows[2]:

$$A = -\frac{x}{y} \tag{61}$$

**Theorem 4.1.** [3] *For the assignment $A$ defined by above formula, $A$ satisfy the flow equation*(25)

*Proof.*

$$da = d(-\frac{x}{y}) = \frac{xdy - ydx}{y^2} = -\frac{dx + ady}{y}$$

and

$$ds = \frac{\sqrt{dx^2 + dy^2}}{y}$$

then

$$\frac{da}{ds} = -\frac{dx + ady}{y} \frac{y}{\sqrt{dx^2 + dy^2}} = -\frac{dx + ady}{\sqrt{dx^2 + dy^2}}$$

Considering that the local coordinate is given by $(-1, 0)$ and $(0, -1)$ under the right-hand rule, we have:

$$\cos\theta = \frac{[dx \quad dy] \begin{bmatrix} \frac{1}{y^2} & 0 \\ 0 & \frac{1}{y^2} \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix}}{\sqrt{[dx \quad dy] \begin{bmatrix} \frac{1}{y^2} & 0 \\ 0 & \frac{1}{y^2} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix}} \sqrt{[-1 \quad 0] \begin{bmatrix} \frac{1}{y^2} & 0 \\ 0 & \frac{1}{y^2} \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix}}}$$

hence

$$\cos\theta = \frac{-dx}{\sqrt{dx^2 + dy^2}}$$

and similarly

[2]This analytic example is provided by Le Zhang, and the geometry interpretation is given by Mingli Yuan
[3]The proof is originally by Le Zhang, and modified by Mingli Yuan

$$\sin\theta = \frac{-dy}{\sqrt{dx^2 + dy^2}}$$

then

$$\frac{da}{ds} = \cos\theta + a\sin\theta$$

□

We can verify $A$ is an eigenfunction of the Laplacian

$$\Delta A = -y^2(\frac{\partial^2}{\partial x^2}A + \frac{\partial^2}{\partial y^2}A) = y^2(\frac{1}{\partial y}(\frac{1}{\partial y}\frac{x}{y})) = 2A$$

### 4.2 A horocycle-based coordinate system

First, we introduce the horocycle-based coordinate system for hyperbolic surfaces. It is a global coordinate system given by two orthogonal sets of circles: one set consists of horocycles that share the same ideal point, while the other consists of geodesics that are perpendicular to the first set.

On the Poincaré disc $\mathcal{P}$, the blue horocycles are tangible at the ideal point $\Omega$, forming the first set of circles. The green lines are geodesics that form the second set. The coordinates of point $P$ are given by the lengths of $OQ$ and $QP$, where point $O$ is the origin and the length is measured using the metric of the hyperbolic surface.

The coordinates of $P$ are $(u, v)$, where the sign of the length is determined by the following rules:

- u: if $P$ is on the same side of $\Omega$, then $u$ is positive; otherwise, $u$ is negative
- v: the sign of the length should satisfy the right-hand rule

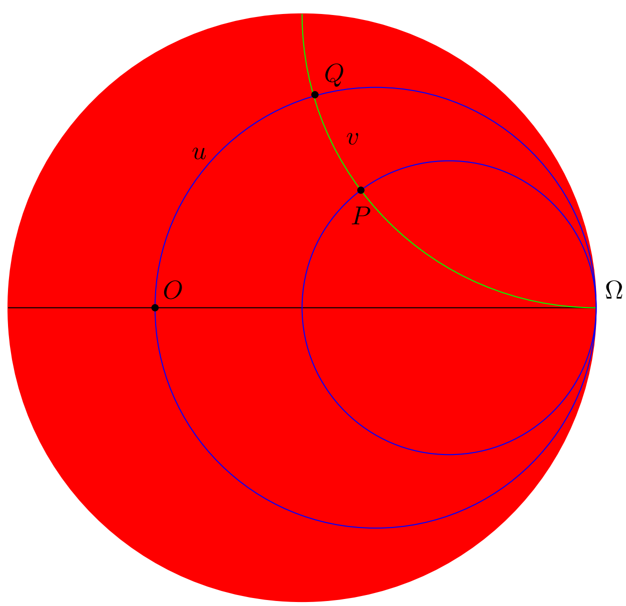


Figure 7: A horocycle-based coordinate system

We can equip it with an inner product

$$\mathbf{a}\cdot\mathbf{b} = \begin{bmatrix} a_u & a_v \end{bmatrix} \begin{bmatrix} e^{-2v} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} b_u \\ b_v \end{bmatrix}$$

and the metrics

$$ds^2 = e^{-2v}du^2 + dv^2$$

Laplacian is given by[1]

$$\Delta = e^{2v}\frac{\partial^2}{\partial u^2} + \frac{\partial^2}{\partial v^2} - \frac{\partial}{\partial v}$$

### 4.3 Example space II

Giving the Poincaré disc $\mathcal{P}$ equipped with the above horocycle-based coordinate system, we consider an assignment function $A$ defined on $\mathcal{P}$ as follows[4]:

$$A = ue^{-v} \tag{62}$$

**Theorem 4.2.** *For the assignment $A$ defined by above formula, $A$ satisfy the flow equation*(25)

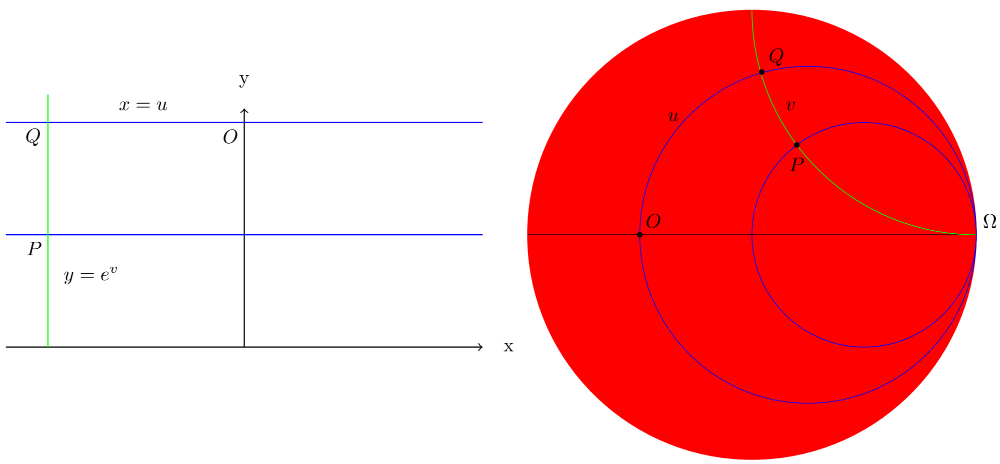


Figure 8: Mapping between two examples

*Proof.* If we introduce complex in the upper half plane model in last section 4.1

$$z = x + yi$$

and stetup a Möbius transform between the upper half plane and current horocycle-based coordinate system:

$$z \mapsto \frac{z-i}{z+i}$$

This transform maps each horizontal lines in $\mathcal{H}$ into the horocycles sharing the same ideal point $\Omega = 1$ in $\mathcal{P}$, also it maps each vertical geodesics in $\mathcal{H}$ into geodesics in $\mathcal{P}$ which are perpendicular to the above horocycles.

And rewrite the Möbius transform in the target coordinate, we get:

$$\begin{cases} x = u \\ y = e^v \end{cases}$$

This lead to

[4]This analytic example is provided by Le Zhang, and the geometry interpretation is given by Mingli Yuan

$$A = -\frac{x}{y} = ue^{-v}$$

And because of theorem **??** and Möbius transform is conformal, we can conclude that $A = ue^{-v}$ obey flow equation.

□

We can verify $A$ is also an eigenfunction of the Laplacian

$$\Delta A = e^{2v}\frac{\partial^2(ue^{-v})}{\partial u^2} + \frac{\partial^2(ue^{-v})}{\partial v^2} - \frac{\partial(ue^{-v})}{\partial v} = 2A$$

From the above proof, we can see that the two assignment function $A$ arose from the same geometry setting, they are equivalent to each other.

### 4.4 Generator independence

Considering the upper half plane $\mathcal{B}$:

$$\{\mathcal{B} : (x, y)|y > 0\}$$

equipped with an inner product and metrics as follows:

$$\mathbf{a} \cdot \mathbf{b} = \begin{bmatrix} a_x & a_y \end{bmatrix} \begin{bmatrix} \frac{1}{\mu^2 y^2} & 0 \\ 0 & \frac{1}{\lambda^2 y^2} \end{bmatrix} \begin{bmatrix} b_x \\ b_y \end{bmatrix}$$

$$ds^2 = \frac{1}{y^2}(\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2})$$

Whatever the choice of $\mu$ and $\lambda$, the assignment is given by

$$A = -\frac{x}{y} \tag{63}$$

We can verify $A$ satisfying the flow equation(25), and also it is generator independent.

**Theorem 4.3.** *The above $A$ satisfying the flow equation*(25)

*Proof.*

$$da = d(-\frac{x}{y}) = \frac{xdy - ydx}{y^2} = -\frac{dx + ady}{y}$$

Notice that

$$ds = \frac{1}{y}\sqrt{\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2}}$$

then

$$\frac{da}{ds} = -\frac{dx + ady}{y} \frac{y}{\sqrt{\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2}}} = \frac{dx + ady}{\sqrt{\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2}}}$$

Consider the local coordinate system given by $(-1, 0)$ and $(0, -1)$ according to the right-hand rule, we have

$$\cos\theta = \frac{\begin{bmatrix} dx & dy \end{bmatrix} \begin{bmatrix} \frac{1}{\mu^2 y^2} & 0 \\ 0 & \frac{1}{\lambda^2 y^2} \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix}}{\sqrt{\begin{bmatrix} dx & dy \end{bmatrix} \begin{bmatrix} \frac{1}{\mu^2 y^2} & 0 \\ 0 & \frac{1}{\lambda^2 y^2} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix}} \sqrt{\begin{bmatrix} -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{1}{\mu^2 y^2} & 0 \\ 0 & \frac{1}{\lambda^2 y^2} \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix}}}$$

Therefore we have

$$\cos\theta = \frac{-\frac{dx}{\mu}}{\sqrt{\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2}}}$$

Similarly

$$\sin\theta = \frac{-\frac{dy}{\lambda}}{\sqrt{\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2}}}$$

Hence we have

$$\frac{da}{ds} = \mu\cos\theta + a\lambda\sin\theta$$

□

### 4.5 Area formula in the global level

In Section 3.4, we introduced an area formula applicable at a local level, as detailed in Equation (51). This section aims to extend that exploration, testing the formula's applicability at a global level.

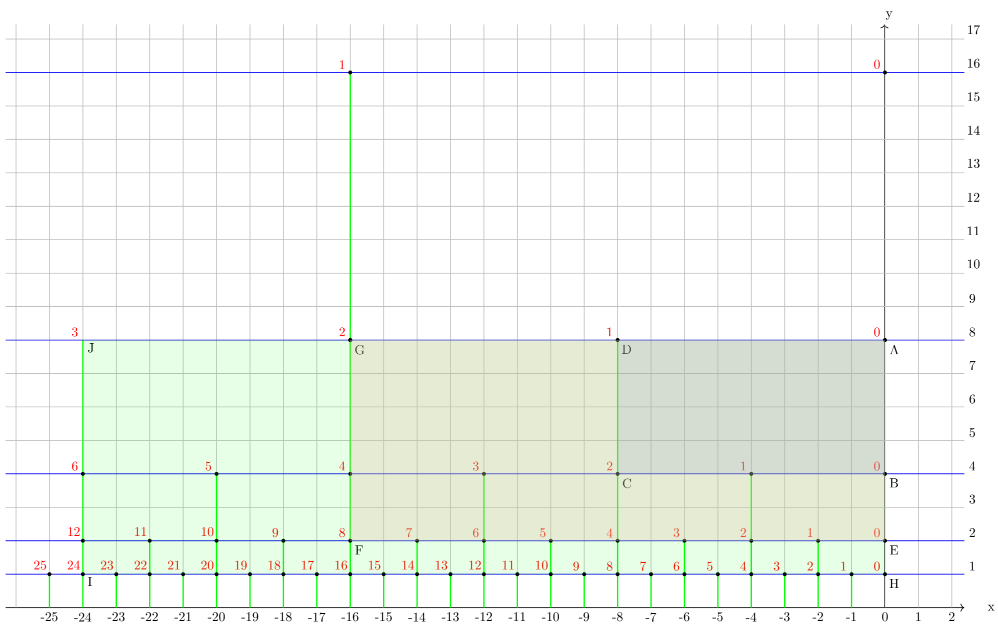


Figure 9: Illustration of the Area Formula

Referencing Figure 9 above, we recall the arithmetic space defined in Section 2.2. As an example, our goal here is to examine the arithmetic torsion across various steps:

For a single step, the scenario unfolds as follows:

$$x \times 2 + 1 - (x + 1) \times 2 = -1 \tag{64}$$

Extending this to two steps, we encounter a different situation:

$$x \times 4 + 2 - (x + 2) \times 4 = -6 \tag{65}$$

And for three steps, the pattern continues:

$$x \times 8 + 3 - (x + 3) \times 8 = -21 \tag{66}$$

The question arises: What is the origin of the numbers $-1$, $-6$, and $-21$? By conceptualizing the area of the quadrilateral $ABCD$ as a basic unit cell, it becomes apparent that the number 1 represents a single unit cell of $ABCD$. Similarly, the number 6 corresponds to the area of $AEFG$, comprising six unit cells; and the number 21 aligns with the area of $AHIJ$, encompassing twenty-one unit cells. This understanding allows us to appreciate how these values emerge naturally from the structure of the arithmetic space, especially we connect area with arithmetic torsion which is defined on edges of that enclosing area.

Now, we proof the above conclusion in a more formal way.

### 4.6 Related problems

*Eigenfunction problem*: TODO.

*Eigenfunction and classification problem*: TODO.

# 5 The second kind arithmetic expression space $\mathfrak{E}_2$

In this section, we delve into an investigation of the fundamental domain within the space $\mathfrak{K}_1$. Our focus encompasses an examination of the discrete symmetry group and an exploration into the methodologies by which symmetry can be achieved through flow.

## 5.1 Fundamental domain in $\mathfrak{K}_1$

## 5.2 Discrete symmetry of $\mathfrak{K}_1$ and its flow realization

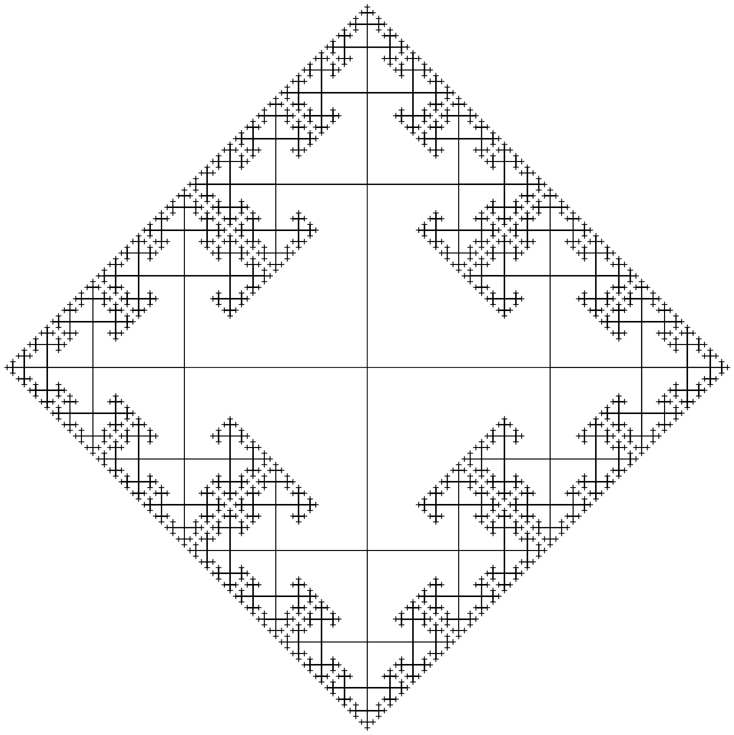

Figure 10: The order-4 Cayley tree

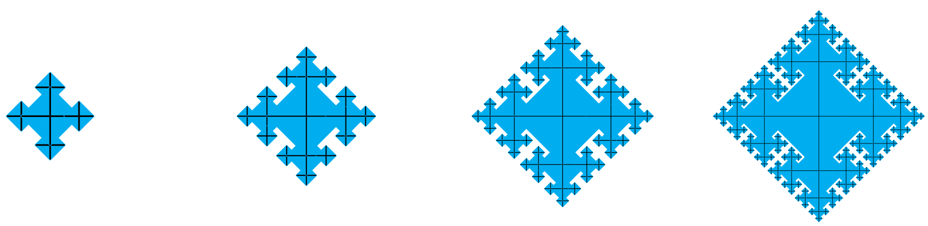

Figure 11: Construction steps 2, 3, 4, and 5 of the tree $T$ and the domain $D$

# 6 On order-4 apeirogonal tiling

In this section, we construct an special domain in the Euclidean plane, and then define a new model for hyperbolic plane based on this domain. We show that this model is isometric to the Poincaré disk model. Finally, we provide two ways to construct the order-4 apeirogonal tiling of the hyperbolic plane. We also defined three scalar fields $X, Y, L$ on the hyperbolic plane which will be used in the next section.

## 6.1 The order-4 Cayley tree $T$

## 6.2 A domain construction based on $T$

Cut and glue

We define a domain $D$ in the Euclidean plane as follows. Let $D$ be the union of the following regions:

### 6.3 $\mathcal{L}_1$ distance and functions $L, X, Y$

### 6.4 Cayley model of the hyperbloic plane $\mathbf{H}_2$

### 6.5 Order-4 apeirogonal tiling

# 7 The third kind of arithmetic expression space $\mathfrak{E}_3$

# 8 Topological arithmetic expression space

## 8.1 The construction of the grid $G_0$

To construct the grid described in subsection **??** and Figure 4, we will use the number theory decomposition introduced by Victor Pambuccian in [5] and Celia Schacht in [8].

$$n = \tau(n)\omega(n)$$

where $\tau(n)$ is a power of 2 and $\omega(n)$ is an odd.

We can see directly from the Figure 4 that the grid is constructed by the following rules:

- horizantal lines (blue, additional lines) satisfying $y = 2^k, k \in \mathbb{Z}$
- vertical lines (green, multiplicative lines) satisfying
  - the value of x satisfying $x = \frac{m}{2^l}, l \in \mathbb{Z}^+, m \in \mathbb{Z}$
  - the assignment begin from $\omega(-m)$ and increase exponentially by power 2.

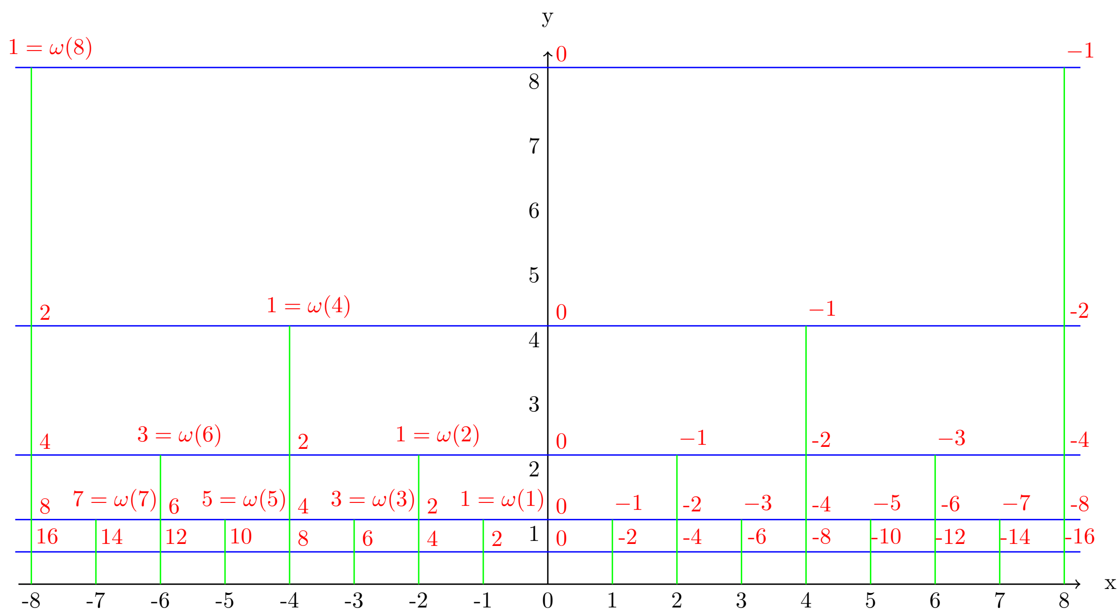

Figure 12: $\omega$ gives the assignment at the start points of vertical lines in $G_0$

The horizontal lines and the vertical lines divide the whole space into a mesh grid $G_0 = (V_0, E_0, F_0)$, where $V_0$ is the set of crossing points, $E_0$ is the set of edges (segments in the horizontal and vertical lines, it should be noted that only the vertical lines are geodesics, while the horizontal lines are horocycles) connecting the crossing points, and $F_0$ is the set of cut cells. This mesh grid is generated by the additional generator 1 and the multiplicative generator 2, and $V_0$, $E_0$, and $F_0$ are all countable sets.

We illustrate the construction schema of $G_0$ in Figure 13.

We notice that $G_0$ is very regular, in fact, all edges are equidistant.

**Lemma 8.1.** *All edges in $G_0$ are equidistant.*

*Proof.* Following the construction schema in Figure 13, we can calculate the length of segments are all equidistant.

Length of $AC$ and $BD$:

$$\int ds = \int_{2^k}^{2^{k+1}} \frac{1}{y \ln 2} dy = \frac{1}{\ln 2}\left(\ln 2^{k+1} - \ln 2^k\right) = 1$$

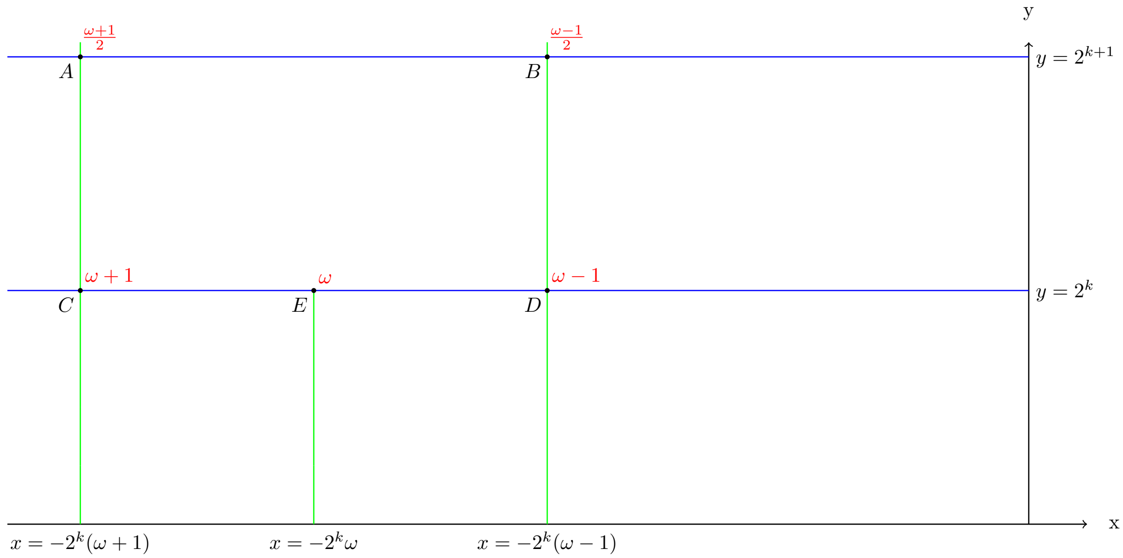


Figure 13: the construction schema of $G_0$

Length of $AB$:

$$\int ds = \int_{-2^k(\omega+1)}^{-2^k(\omega-1)} \frac{1}{2^{k+1}} dx = \frac{1}{2^{k+1}} \left(2^{k+1}\right) = 1$$

Length of $CE$:

$$\int ds = \int_{-2^k(\omega+1)}^{-2^k\omega} \frac{1}{2^k} dx = \frac{1}{2^k} \left(2^k\right) = 1$$

Length of $ED$:

$$\int ds = \int_{-2^k\omega}^{-2^k(\omega-1)} \frac{1}{2^k} dx = \frac{1}{2^k} \left(2^k\right) = 1$$

□

### 8.2 The construction of the grid $G_1$

We can similarly construct the grid $G_1$ using the additional generator $\frac{1}{2}$ and the multiplicative generator $\sqrt{2}$, the grid $G_2$ using the additional generator $\frac{1}{4}$ and the multiplicative generator $\sqrt[4]{2}$, and so on. And each time the cell of the mesh grid is divided into smaller cells and the end points of the vertical lines move upward.

### 8.3 The construction of the grid $G_2$

### 8.4 The grid mesh is dense

It is easy to see that there is a chain of inclusion relations:

$$V_0 \subset V_1 \subset V_2 \subset \cdots V_i \subset \cdots$$

Suppose $V = \bigcup_{i=1}^{\infty} V_i$, we have below lemma

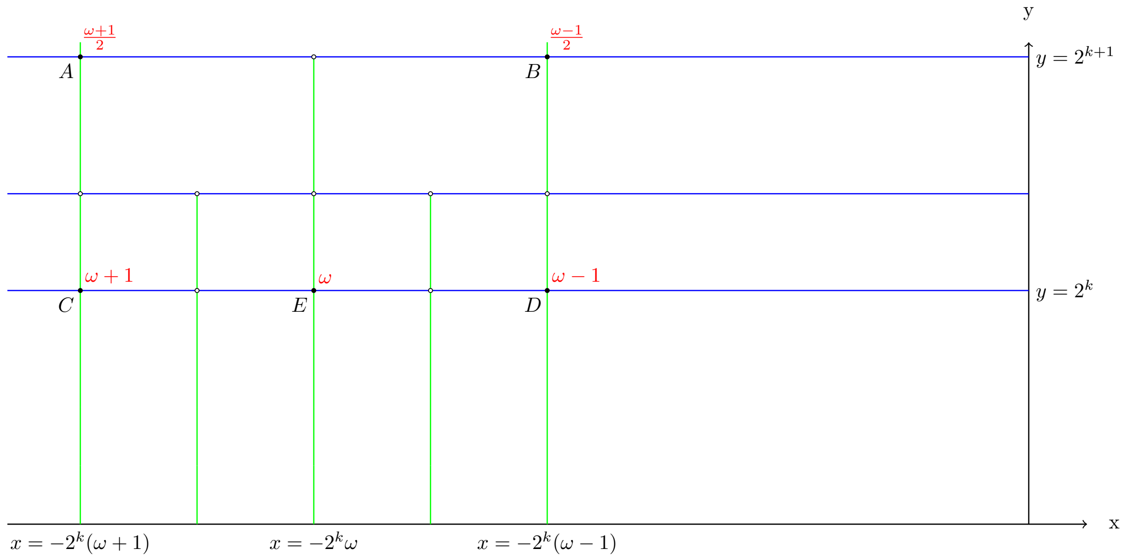


Figure 14: the construction schema of $G_1$

**Lemma 8.2.** *$V$ is a countable dense set.*

*Proof.* Because $V_i$ is countable, and the union is over a countable index set, so $V$ is countable.

We can prove it is dense by contradiction. Suppose $V$ is not a dense set. Then there is a point $p$ in the space neither belongs to $V$ nor is a limit point of $V$.

TODO...

□

### 8.5 Completeness and topology

### 8.6 As a special integral

# 9 From arithmetic torsion to curvature

## 10 Tube structure, complexification and fibration

# 11 General discussion

## 11.1 On integral theory

## 11.2 On limitation and boundary

## 11.3 On representation of function

## 11.4 Questions related with complexity?

# 12 A glossary of unsolved problems

## 12.1 Foundation questions

## 12.2 Classification problem

## 12.3 Eigenfunction problem

## 12.4 Tube structure

## 12.5 Singular points and divergent series

## 12.6 Function and a new calculus?

## 12.7 Category of function theories?

## 12.8 Computation geometry

## 12.9 Logic geometry

### 12.9.1 irrationality of $\sqrt{17}$

## References

[1] S. S. e Costa. A description of several coordinate systems for hyperbolic spaces. *arXiv: Mathematical Physics*, 2001.

[2] Donald Ervin Knuth. The art of computer programming, volume i: Fundamental algorithms, 2nd edition. 1997.

[3] Peter W. Markstein. Software division and square root using goldschmidt's algorithms. 2004.

[4] Tristan Needham. Visual differential geometry and forms. 2021.

[5] Victor Pambuccian. The arithmetic of the even and the odd. *The Review of Symbolic Logic*, 9:359 – 369, 2016.

[6] Eric Quinnell, Earl E. Swartzlander, and Carl Lemonds. Floating-point fused multiply-add architectures. *2007 Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*, pages 331–337, 2007.

[7] John C. Reynolds. Definitional interpreters for higher-order programming languages. *Higher-Order and Symbolic Computation*, 11:363–397, 1972.

[8] Celia Schacht. Another arithmetic of the even and the odd. *The Review of Symbolic Logic*, 11:604 – 608, 2018.

## A Solution of the flow equation

We can also get a direct formal solution of the flow equation ((25)) step by step:

$$\frac{da}{\mu\cos\theta + a\lambda\sin\theta} = ds$$

$$\frac{1}{\lambda\sin\theta}\frac{d(\mu\cos\theta + a\lambda\sin\theta)}{\mu\cos\theta + a\lambda\sin\theta} = ds$$

$$\frac{1}{\lambda\sin\theta} ln(\mu\cos\theta + a\lambda\sin\theta) = s + C$$

$$\mu\cos\theta + a\lambda\sin\theta = e^{\lambda s\sin\theta}e^{C\lambda\sin\theta}$$

Considering the initial condition

$$\mu\cos\theta + a_0\lambda\sin\theta = e^{C\lambda\sin\theta}$$

We have

$$\mu\cos\theta + a\lambda\sin\theta = e^{\lambda s\sin\theta}(\mu\cos\theta + a_0\lambda\sin\theta)$$

$$a = \frac{\mu\cos\theta + a_0\lambda\sin\theta}{\lambda\sin\theta}e^{\lambda s\sin\theta} - \frac{\mu}{\lambda}\cot\theta$$

$$a = (a_0 + \frac{\mu}{\lambda}\cot\theta)e^{\lambda s\sin\theta} - \frac{\mu}{\lambda}\cot\theta$$

$$a = a_0 e^{\lambda s\sin\theta} + \frac{\mu}{\lambda}(e^{\lambda s\sin\theta} - 1)\cot\theta \tag{67}$$

$$a = a_0 e^{\lambda s\sin\theta} + \frac{\mu}{\lambda}(e^{\lambda s\sin\theta} - 1)\cot\theta \tag{68}$$

## B Infinitesimal-discrete conformance

In order to verify the conformance, we expand the formula (68) in the following way:

$$a = a_0 e^{\lambda s \sin\theta} + \frac{\mu}{\lambda}[1 + \lambda s \sin\theta + \frac{1}{2!}(\lambda s \sin\theta)^2 + \frac{1}{3!}(\lambda s \sin\theta)^3 + \cdots - 1]\cot\theta \tag{69}$$

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta + \frac{\mu}{\lambda}\sin\theta\cos\theta(\frac{\lambda^2 s^2}{2!} + \frac{\lambda^3 s^3}{3!}\sin\theta + \frac{\lambda^4 s^4}{4!}\sin^2\theta + \cdots) \tag{70}$$

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta + \frac{\mu}{2\lambda}\sin 2\theta(\frac{\lambda^2 s^2}{2!} + \frac{\lambda^3 s^3}{3!}\sin\theta + \frac{\lambda^4 s^4}{4!}\sin^2\theta + \cdots) \tag{71}$$

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta + \frac{\mu}{2\lambda}\Psi(s)\sin 2\theta \tag{72}$$

When $\theta = \frac{k\pi}{2}, k = 0, 1, 2, 3\cdots, s = 0, 1, 2, 3\cdots$, we have

$$a = a_0 e^{\lambda s \sin\theta} + \mu s \cos\theta \tag{73}$$

Especially, we have

$$a = a_0 + \mu s, s = 0, 1, 2, 3\cdots, k = 0, 1, 2, 3\cdots, \theta = 2k\pi \tag{74}$$

$$a = x_0 e^{\lambda s}, s = 0, 1, 2, 3\cdots, k = 0, 1, 2, 3\cdots, \theta = 2k\pi + \frac{\pi}{2} \tag{75}$$

$$a = a_0 - \mu s, s = 0, 1, 2, 3\cdots, k = 0, 1, 2, 3\cdots, \theta = 2k\pi + \pi \tag{76}$$

$$a = a_0 e^{-\lambda s}, s = 0, 1, 2, 3\cdots, k = 0, 1, 2, 3\cdots, \theta = 2k\pi + \frac{3\pi}{2} \tag{77}$$

which gives the conformance.

# C Geometry calculation

Giving

$$ds^2 = \frac{1}{y^2}(\frac{dx^2}{\mu^2} + \frac{dy^2}{\lambda^2})$$

we calculate the geometric quantities. We follow the notion in the text book[4], and the first fundamental form is given by

$$ds^2 = A^2dx^2 + B^2dy^2$$

where

$$A = \frac{1}{\mu y}, \quad B = \frac{1}{\lambda y}$$

## C.1 Line element

The line element is already given by above equation.

## C.2 Area element

The area element is given by

$$dS = ABdxdy$$

hence we have

$$dS = \frac{1}{\mu\lambda y^2}dxdy$$

## C.3 Gauss curvature

Gauss curvature $K$ is given by

$$K = -\frac{1}{AB}\left(\partial_y\left(\frac{\partial_y A}{B}\right) + \partial_x\left(\frac{\partial_x B}{A}\right)\right)$$

so we have

$$K = -\mu\lambda y^2\left(\partial_y\left(\lambda y\partial_y\left(\frac{1}{\mu y}\right)\right) + \partial_x\left(\mu y\partial_x\left(\frac{1}{\lambda y}\right)\right)\right)$$

$$K = -\lambda^2 y^2\left(\partial_y\left(y\partial_y\left(\frac{1}{y}\right)\right)\right)$$

$$K = -\lambda^2 y^2\frac{1}{y^2}$$

$$K = -\lambda^2$$

### C.4 Laplacian

Given a metric tensor

$$g = \begin{bmatrix} A^2 & 0 \\ 0 & B^2 \end{bmatrix},$$

where $A$ and $B$ are functions of the coordinates (typically $x$ and $y$), the Laplacian of a function $f(x, y)$ can be derived from the general expression of the Laplace-Beltrami operator for a Riemannian manifold. The formula for the Laplacian $\Delta f$ in such a setting, using the metric components $g_{ij}$, is given by:

$$\Delta f = \frac{1}{\sqrt{|g|}} \partial_i \left( \sqrt{|g|} g^{ij} \partial_j f \right),$$

where $|g|$ is the determinant of the metric tensor $g_{ij}$, $g^{ij}$ are the components of the inverse metric tensor, and $\partial_i$ denotes partial differentiation with respect to the $i$th coordinate.

Given the metric tensor, the determinant $|g|$ is $A^2 B^2$. The inverse metric tensor $g^{ij}$ is simply:

$$g^{ij} = \begin{bmatrix} \frac{1}{A^2} & 0 \\ 0 & \frac{1}{B^2} \end{bmatrix}.$$

Plugging these into the formula for the Laplacian, we get:

$$\Delta f = \frac{1}{AB} \left[ \partial_x \left( BA^{-1} \partial_x f \right) + \partial_y \left( AB^{-1} \partial_y f \right) \right],$$

In our setting, $A = \frac{1}{\mu y}$ and $B = \frac{1}{\lambda y}$:

$$\Delta f = y^2 \left( \mu^2 \frac{\partial^2 f}{\partial x^2} + \lambda^2 \frac{\partial^2 f}{\partial y^2} \right)$$

And for the function $f = -\frac{x}{y}$, we have

$$\Delta f = -\frac{2\lambda^2 x}{y} = 2\lambda^2 f$$

So, we reach the conclusion that the function $f = -\frac{x}{y}$ is a eigenfunction of the Laplacian with eigenvalue $2\lambda^2$.

## D Grid calculation

### D.1 Arc length

### D.2 Perimeter

### D.3 Area

### D.4 Arithmetic torsion

Arithmetic torsion integrated over a unit cell $U$ is given by

$$\int_U \tau \, dS$$

# E Arithmetic expression, combinators and transformation over trees

## E.1 LISP and combinators

## E.2 Applicative and concatenative

## E.3 Donaghey transformation