The Continuum Hypothesis

Chris Pindsle

7 March 2018

Abstract

A proof of the Continuum Hypothesis as originally posed by Georg Cantor in 1878; that an uncountable set of real numbers has the same cardinality as the set of all real numbers. Any set of real numbers can be encoded by the infinite paths of a binary tree. If the binary tree has an uncountable node it must have a descendant with 2 uncountable successors. Each of those will have descendants with 2 uncountable successors, recursively. As a result the infinite paths of an uncountable binary tree will have the same cardinality as the set of all real numbers, as will the uncountable set of real numbers encoded by the tree.

1 The Hypothesis

We know that some sets are countable, ie they can be put into a bijection with $\mathbb{N}$. There are other sets where this can't be done, eg an interval of $\mathbb{R}$. We want to prove that there are no sets whose cardinality lies in between. That is, if a set is not countable its cardinality is at least that of $\mathbb{R}$.

2 Outline of the Proof

I intend to prove the hypothesis in its original form as proposed by Georg Cantor in 1878[1]:

Any uncountable set of real numbers is equinumerous with $\mathbb{R}$

Since there is a bijection between the open interval (0,1) and the set of all the real numbers, there is a bijection between any subset of (0,1) and a subset of $\mathbb{R}$. Therefore it is sufficient to prove:

Any uncountable subset of (0,1) is equinumerous with $\mathbb{R}$

2.1 Representation using Binary Trees

Any real number in (0,1) can be expressed using infinite binary notation, with a leading 0 before the binary point and an infinite string of binary digits (bits) after the point. Therefore any subset of (0,1) can represented by an infinite binary tree with 0 as the root node.

[1] "The conjecture that any uncountable set of real numbers is equinumerous with $\mathbb{R}$ was first posed by Cantor (1878), and it is the first version of what is called the continuum hypothesis." Stillwell, *The Real Numbers*, p79

Since the binary tree represents real numbers it is convenient not to have terminating nodes. Then each node has either 1 or 2 immediate successors. Rational numbers which would otherwise terminate are continued with an infinite string of repeating 0's.

So for convenience I am using the following definition of a binary tree:

Definition. A **binary tree** is a rooted tree where every node has one or two successors

Which means that in this document the term *binary tree* actually refers to an infinite binary tree. Using binary trees to represent real numbers has some ambiguity in that there may be more than one way of representing some of the numbers. In particular, fractions whose denominators are a power of two can be represented using either trailing 0's or 1's.

However any string from a binary tree specifies a particular real number.

2.2 Top Level of the Proof

1. Any subset of $\mathbb{R}$ has a bijection with the infinite paths of a binary tree
2. If the subset of $\mathbb{R}$ is uncountable then the binary tree must also be uncountable
3. An uncountable binary tree contains a tree isomorphic to a complete binary tree
4. Since a complete binary tree has cardinality $\mathfrak{c}$, therefore the uncountable binary tree and also the subset of $\mathbb{R}$ must have cardinality $\mathfrak{c}$

2.3 Uncountable Binary Trees

We can prove that if a node of a binary tree is uncountable it must have a descendant node with two uncountable successor nodes.

Assume the converse, ie that we have an uncountable node T_0 where every descendant has at most one uncountable successor. However T_0 must have at least one uncountable successor otherwise it is countable itself. Therefore it must have exactly one uncountable successor, say T_1.

We can apply the same argument to T_1 implying that it has exactly one uncountable subtree T_2. In this way we form a denumerable chain of uncountable subtrees $T_0, T_1, T_2, ...T_n, ...$ Each node T_n in the chain has the property that it has exactly one uncountable successor T_{n+1}. They each may have either zero or one countable subtrees.

As a consequence and contrary to assumption T_0 is countable:

- The chain of nodes $T_0, T_1, T_2, ...T_n, ...$ defines one number
- The remaining numbers belong to one of the countable subtrees and so are countable since a countable union of countable sets is countable.

Therefore an uncountable tree must have a descendant node which has two uncountable subtrees. Applying the same argument recursively to each of those subtrees we obtain a binary tree with an isomorphism to a complete infinite binary tree.

Since the infinite paths of a complete infinite binary tree are equinumerous with $\mathbb{R}$ it implies that the infinite paths of an uncountable binary tree are also equinumerous with $\mathbb{R}$.

3 Detailed Proof

3.1 A Subset of $\mathbb{R} \Leftrightarrow$ Some Subset of (0,1)

Lemma 3.1.1. *Any subset of $\mathbb{R}$ has a bijection with some subset of (0,1)*

Proof. For example let $\varphi : \mathbb{R} \mapsto (0,1)$ be defined by $\varphi(x) = \frac{1}{1+e^x}$. Then:

- $\varphi(x_1) = \varphi(x2) \Rightarrow x_1 = x_2, \forall x_1, x_2 \in \mathbb{R}$ since $\frac{1}{1+e^{x_1}} = \frac{1}{1+e^{x_2}} \Rightarrow \log(e^{x_1}) = \log(e^{x_2})$ and $\exp(x) : (-\infty, \infty) \mapsto (0, \infty)$ while $\log(x) : (0, \infty) \mapsto (-\infty, \infty)$. Therefore φ is injective
- If $\varphi(x) = \frac{1}{1+e^x} = y$ then $x = \log(\frac{1}{y} - 1)\ \forall y : 0 < y < 1$ That is, for any $y \in (0,1)$ we can find an x such that $\varphi(x) = y$. So φ is also surjective

Therefore φ defines a bijection between $\mathbb{R}$ and (0,1). □

3.2 A Subset of (0,1) $\Leftrightarrow$ Some Binary Tree

Definition. denumerable means countably infinite

Definition. SS is the set of strings with a leading 0, followed by a denumerable number of 0's and 1's

Lemma 3.2.1. *For every number r in (0,1) there is a string $S \in SS$ that defines a series whose limit equals r*

We are using the normal method of encoding an infinite binary number. The value of an infinite binary string after n places is given by $0.b_1b_2...b_n$ where a bit b_n with a value of 1 adds $\frac{1}{2^n}$

Proof. Let $S_n = 0.b_1b_2...b_n$ be the value of a string with n binary places.

Given any $r \in (0,1)$ assume we have constructed S_n so that $S_n < r < S_n + \frac{1}{2^n}$

In order to calculate the next bit:

If $r > S_n + \frac{1}{2^{n+1}}$ then take the value of b_{n+1} as 1

after which $S_{n+1} = S_n + \frac{1}{2^{n+1}}$, so now

$S_n + \frac{1}{2^{n+1}} < r < S_n + \frac{1}{2^n}$, ie

$S_{n+1} < r < S_{n+1} + \frac{1}{2^{n+1}}$ since $S_n + \frac{1}{2^n} = S_n + \frac{2}{2^{n+1}} = S_{n+1} + \frac{1}{2^{n+1}}$

on the other hand:

if $r < S_n + \frac{1}{2^{n+1}}$ then take the value of b_{n+1} as 0

after which $S_{n+1} = S_n$, so still

$S_{n+1} < r < S_{n+1} + \frac{1}{2^{n+1}}$

of course if $r = S_n + \frac{1}{2^{n+1}}$ then the process terminates; we just set b_{n+1} to 1 and fill the rest of the string with an infinite sequence of zeros.

If we take S_0 to be 0 then since $0 < r < 1 (= 1/2^0)$ we can apply the same procedure to find b_1. So for any series $S_n = 0.b_1 b_2 ... b_n$ built using this procedure, we can always choose a b_{n+1} so that:

$|r - S_{n+1}| < \frac{1}{2^{n+1}}$,

which has limit zero as we increase n by adding digits to the infinite binary. Therefore the string $S = 0.b_1 b_2 ... b_n ...$ defines a series whose limit equals r. □

3.2.1 Defining a bijection between (0,1) and a subset of SS

The map from strings in SS to numbers in (0,1) is not injective because more than one string can define the same number. However the strings in SS belong to equivalence classes where strings in the same class define the same number. We can define a bijective map from a subset of the equivalence classes of SS to numbers in (0,1):

- Strings in different classes map to different numbers, so the map is injective
- Lemma 3.2.1 shows that every number r in (0,1) is defined by some string in SS. Therefore one of the equivalence classes will map to r, so the map is also surjective

In order to make a bijective map using the individual strings we need to select one from each of the relevant equivalence classes.

Lemma 3.2.2. *There is a bijective map between a subset SR* of the open interval (0,1) and some subset SS* of SS*

Note: SS* is always a proper subset of SS because the full set of strings in SS includes those that define 0 and 1.

Definition. Strings in SS belong to the same equivalence class if they define the same number

Proof. Let SS* be a subset of SS that has one string from each equivalence class that maps to a number in SR*.

Let $\varphi : SS^* \mapsto SR^*$ be the map that takes each string in SS* and maps it to the corresponding number in SR*

From lemma 3.2.1 we know that every number in (0,1) is defined by some string in SS, therefore there will be a string in SS* to map onto every number in SR*. That is, φ is surjective.

Since SS* only takes a single string from each equivalence class, and strings from different equivalence classes define different numbers, it follows that each different string in SS* maps to a different number in SR*. That is, φ is injective. Therefore φ is a bijective map from SS* to SR*. □

It is convenient to use the following term for an infinite path from the root node.

Definition. A **strand** is an infinite path from the root of an infinite binary tree

Lemma 3.2.3. *Given a subset SR* of (0,1), there is a bijection between SR* and the strands of some binary tree BT*

Strings in SS can be put into a tree where those strings that share the first n bits will share a node at that depth in the tree. Since every string in SS has a leading zero followed by a denumerable string of zeros and ones, the root node will be a zero and it will be an infinite binary tree.

Proof. From lemma 3.2.2 there is a bijection between SR* and a subset SS* of SS. Let BT be the binary tree that is created from the strings in SS*. Let $\varphi : SS^* \mapsto BT$ take strings in SS* to the corresponding strand in BT:

- φ is surjective because every string in SS* is included in BT
- φ is injective because different strings in SS* will map to strands in BT that differ after a certain number of bits

Therefore φ is bijective. From lemma 3.2.2 there is a bijection between SR* and SS*, therefore there is a bijection between SR* and the strands of BT. □

Since we already know from lemma 3.1.1 that any set of real numbers has a bijection with a subset of (0,1) this result means that any subset of $\mathbb{R}$ has a bijection with the strands of a binary tree.

3.3 If a Subset of $\mathbb{R}$ is Uncountable its Binary Tree must also be Uncountable

Definition. The **cardinality** of a binary tree is determined by the cardinality of its strands

Lemma 3.3.1. *If a subset of* $\mathbb{R}$ *is uncountable it has a bijection with the strands of an uncountable binary tree*

Proof. Lemma 3.1.1 shows that any subset of $\mathbb{R}$ has a bijection with some subset of (0,1) while lemma 3.2.3 shows that a subset of (0,1) has a bijection with the strands of some binary tree. Combined they prove that there is a bijection between the numbers in any subset of $\mathbb{R}$ and the strands of some binary tree. Therefore they have the same cardinality so if the subset of $\mathbb{R}$ is uncountable then the strands of its binary tree must also be uncountable. □

3.4 An Uncountable Binary Tree has a Descendant that has Two Uncountable Successors

Definition. A **node(i)** is a set of strands that share the first i binary digits. $i \in \mathbb{N}, i \geq 1$

Note: i = 1 means they just share the leading zero before the binary point.

Definition. A **descendant** of a node(i) is a node which is a subset of node(i)

Note: Since the subset is not necessarily a proper subset, the definition means that a node is a descendant of itself.

Definition. A **successor** of node(i) is a descendant of node(i) and is a node(i+1)

Note: This definition means that a node cannot be its own successor.

Definition. The strands of a **countable node** can be put into a 1:1 relationship with a subset of $\mathbb{N}$

Note: Since the subset of $\mathbb{N}$ is not necessarily a finite subset the definition includes countably infinite nodes.

Lemma 3.4.1. *A union of countable sets is countable*

Proof omitted: For example refer Freiwald, *An Introduction to Set Theory and Topology*, p30

Lemma 3.4.2. *If T_n is an uncountable node of an infinite binary tree, with at most one uncountable successor then T_n has exactly one uncountable successor*

Proof. If T_n has no uncountable successors then each of its successors must be countable, in which case T_n would also be countable by lemma 3.4.1. So it must have at least one uncountable successor.

By assumption T_n has at most one uncountable successor, therefore it must have exactly one uncountable successor. □

Lemma 3.4.3. *If T_0 is a node which is uncountable and all of its descendants have at most one uncountable successor then there is a chain of uncountable nodes $T_0, T_1, T_2, ...T_n, ...$ where each T_n has the property that T_{n+1} is its only uncountable successor*

Proof. Since T_0's descendants include itself, it is also assumed to have at most one uncountable successor. Therefore since it is uncountable, it has exactly one uncountable successor T_1 by lemma 3.4.2

If T_k is a descendant of T_0 then all of T_k's descendants must also have at most one uncountable successor. Therefore if T_k is an uncountable descendant of T_0 it has exactly one uncountable successor T_{k+1} also by lemma 3.4.2

The lemma follows by induction. □

Theorem 1. *An uncountable node in an infinite binary tree must have a descendant that has two uncountable successors.*

Proof. Assume there is a node T_0 which is uncountable and that all of its descendants have at most one uncountable successor. Then by lemma 3.4.3 there is a chain of uncountable nodes $T_0, T_1, T_2, ...T_n, ...$ where each T_n has the property that T_{n+1} is its only uncountable successor. Since the chain must be a subset of one of the strings in SS from which the tree was created, it's nodes must be denumerable.

In which case there is a map from the strands in T_0 to $\mathbb{N}$:

- Map the strand defined by the chain $T_0, T_1, T_2, ...T_n, ...$ to 1
- The remaining strands in T_0 are the countable successors of those nodes in $T_0, T_1, T_2, ...T_n, ...$ that have a countable as well as an uncountable successor. They form a countable union of countable sets (since there are a denumerable number of nodes in the chain), so by lemma 3.4.1 it is possible to map them to the remaining numbers in $\mathbb{N}$, starting with 2

So if there is an uncountable node where all of its descendants have at most one uncountable successor, then contrary to assumption the node is countable. It follows that every uncountable node must have a descendant with two uncountable successors. □

An explicit map from T_0 to $\mathbb{N}$

The strands from the countable successors in the chain $T_0, T_1, T_2, ...T_n, ...$ can be mapped to $\mathbb{N}$ using a similar diagonal method to that used for $\mathbb{Q}$.

In the following table the rows denote the junctions that have countable successors. The columns denote the strands from each of those successors. The numbers in the cells show the order of counting. It assumes that the countable successors are all countably infinite.

	1	2	3	...	m	...
1	1	2	4	...	$\frac{m(m-1)}{2}+1$	...
2	3	5	...		$\frac{(m+1)m}{2}+2$	...
3	6	...			$\frac{(m+2)(m+1)}{2}+3$	...
...	...				...	
n	$\frac{n(n+1)}{2}$	...			$\frac{(m+n-1)(m+n-2)}{2}+n$	...
...	...				...	

After including the initial strand from the chain $T_0, T_1, T_2, ...T_n, ...$, the $\mathrm{m^{th}}$ strand from the countable successor of the $\mathrm{n^{th}}$ junction is mapped to $\frac{(m+n-1)(m+n-2)}{2}+n+1$.

It follows that every strand in the tree defined by T_0 is eventually counted.

3.5 An Uncountable Binary Tree is Equinumerous with $\mathbb{R}$

We know from theorem 1 that any uncountable node has a descendant node with two uncountable successors. Applying this recursively we get a tree with the same structure as a complete binary tree. We can call it an isocomplete binary tree. The strands that connect the junctions of the isocomplete binary tree should be equinumerous with those of a complete binary tree, which has the same cardinality as $\mathbb{R}$. Since the strands in the isocomplete binary tree are a subset of the strands in the uncountable tree, it follows that the uncountable tree's strands are at least equinumerous with $\mathbb{R}$.

To prove this we first show that the real junctions of an uncountable binary tree have the same relationship of descent as the nodes of a complete binary tree. We can then show that the real junctions induce strands in the uncountable tree that are equinumerous with those of a complete binary tree.

Definition. A **junction** is a node with two successors

Definition. A **real junction** is a node with two uncountable successors

Definition. Every node of a **complete binary tree** has two successor nodes

3.5.1 Real Junctions $\Leftrightarrow$ Nodes of a Complete Binary Tree

Definition. $a \succ b$ means a is an ancestor of b

Definition. Λ is a function that maps nodes to their label, either 0 or 1

Definition. The **real junction root** of an uncountable binary tree is the first descendant of the root node of the tree that has two uncountable successors

Definition. A real junction r_j is the **nearest real descendant** of a node n if there is no real junction r_i such that $n \succ r_i \succ r_j$

Lemma 3.5.1. *An uncountable binary tree has a unique real junction root*

Proof. By theorem 1 an uncountable tree must have a node with two uncountable successors, so there must be a real junction descended from the root of the tree. If each of the root node's successors has a descendant with 2 uncountable successors then the root node itself is the real junction root. Otherwise the root node's nearest real descendant is the real junction root. □

Lemma 3.5.2. *There is an isomorphism between the real junctions of an uncountable binary tree and the nodes of a complete binary tree which preserves labels and the relation of descent*

We will prove the lemma by defining a binary tree **RJT** whose nodes are mapped isomorphically to the real junctions of an uncountable binary tree **UNC**. We will map the real junction root r_0 of UNC to the root c_0 of RJT.

Let **RJ** be the set of real junctions in UNC. To prove the isomorphism we prove that there is a bijective map $\varphi : RJ \mapsto nodes(RJT)$, with the properties that when $\varphi(r_0) \mapsto c_0$:

- $r_i \succ r_j \Leftrightarrow \varphi(r_i) \succ \varphi(r_j)$ and
- $\Lambda(r) = \Lambda(\varphi(r))$

where r_i, r_j and r are real junctions in RJ

We use induction to establish the isomorphism. We also need to prove that RJT is a complete binary tree.

Proof. By lemma 3.5.1 UNC must have real junction root $r_0 \in RJ$. Label it with 0 and map it to the root of RJT. That is, $\varphi(r_0) = c_0$ where c_0 is the root of RJT. Also label c_0 with 0. So $\Lambda(r_0) = \Lambda(\varphi(r_0)) = 0$. Define them both as being at level 1.

By theorem 1, every real junction in UNC has two real junctions as descendants. So every real junction in UNC has two successors labelled 0 and 1, which are the ancestors of the two nearest real descendants. Label them 0 or 1 corresponding to which successor they are descended from. (This includes the possibility that either descendant in UNC is also the successor in UNC.)

Inductive step:

Assume that there is an isomorphism between the respective real junctions of RJ and nodes of RJT for all levels from 1 to k. That is, there is a map $\varphi : RJ \mapsto RJT$ where:

- $r_1 \succ r_2 \Leftrightarrow \varphi(r_1) \succ \varphi(r_2)$ for all real junctions $r_1, r_2 \in RJ$ in levels 1 to k
- $\Lambda(r) = \Lambda(\varphi(r))$ for all real junctions $r \in RJ$ in levels 1 to k

For every real junction r_k in RJ at level k and corresponding node $c_k = \varphi(r_k)$ in RJT at level k, we extend the isomorphism by mapping each of r_k's nearest real descendants, say r_{k0} and r_{k1} labelled 0 and 1 to corresponding successors of c_k say c_{k0} and c_{k1}. That means $\varphi(r_{k0}) = c_{k0}$ and $\varphi(r_{k1}) = c_{k1}$ and also that $\Lambda(r_{k0}) = \Lambda(c_{k0})$ and $\Lambda(r_{k1}) = \Lambda(c_{k1})$

Since r_{k0} and r_{k1} are descendants of r_k, then for any real junction $r \in RJ$ in levels 0 to k, $r \succ r_{k0}$ and $r \succ r_{k1}$ if and only if $r \succ r_k$. We can summarise this as $r \succ r_{k+1}$ if and only if $r \succ r_k$.

The same way $\varphi(r) \succ \varphi(r_{k+1})$ if and only if $\varphi(r) \succ \varphi(r_k)$ for any node $\varphi(r) \in RJT$ in levels 0 to k.

We already have $r_1 \succ r_2 \Leftrightarrow \varphi(r_1) \succ \varphi(r_2)$ for all real junctions $r_1, r_2 \in RJ$ in levels 0 to k. In particular $r \succ r_k \Leftrightarrow \varphi(r) \succ \varphi(r_k)$. Therefore:

> $r \succ r_{k+1} \Leftrightarrow r \succ r_k \Leftrightarrow \varphi(r) \succ \varphi(r_k) \Leftrightarrow \varphi(r) \succ \varphi(r_{k+1})$ for any real junctions $r \in RJ$ in levels 0 to k, any r_k at level k and r_{k+1} at level k+1.

As a result:

> $r_1 \succ r_2 \Leftrightarrow \varphi(r_1) \succ \varphi(r_2)$ for all real junctions $r_1, r_2 \in RJ$ in levels 0 to k+1

The inductive step for the label:

$\Lambda(r_{k0}) = \Lambda(c_{k0}) = \Lambda(\varphi(r_{k0}))$ and $\Lambda(r_{k1}) = \Lambda(c_{k1}) = \Lambda(\varphi(r_{k1}))$ Therefore since $\Lambda(r) = \Lambda(\varphi(r))$ for all real junctions $r \in RJ$ in levels 0 to k, it follows that:

> $\Lambda(r) = \Lambda(\varphi(r))$ for all real junctions $r \in RJ$ in levels 0 to k+1

The root node:

We have that $\varphi : r_0 \mapsto c_0$ so $r_0 \succ r_0 \Rightarrow \varphi(r_0) \succ \varphi(r_0)$

Additionally $\Lambda(r_0) = 0$ and $\Lambda(c_0) = 0$ so $\Lambda(r_0) = \varphi(\Lambda(r_0))$.

It follows that the isomorphism between the real junctions in RJ and nodes in RJT extends to all levels 0,1,2, ... n, ...

$\varphi : RJ \mapsto RJT$ is bijective:

It is injective because:

> $\varphi(r_1) = \varphi(r_2) \Rightarrow (\varphi(r_1) \succ \varphi(r_2)$ and $\varphi(r_2) \succ \varphi(r_1)) \Rightarrow (r_1 \succ r_2$ and $r_2 \succ r_1) \Rightarrow r_1 = r_2$

φ is also surjective since nodes are only defined in RJT as the result of being the target of some $\varphi(r)$.

RJT is a complete binary tree:

Since each real junction in RJ has two nearest real descendants and $\varphi : RJ \mapsto RJT$ is an isomorphism that preserves descent, it follows that every node of RJT has 2 successors. □

3.5.2 Rooted Real Junction Sequences ⇔ Strands of a Complete Binary Tree

Since it is the real junctions of an uncountable tree that have an isomorphism with the nodes of a complete binary tree, we first show that there is a bijection between an infinite sequence of those junctions and the strands of a complete binary tree. Later we show that the real junction sequences also have a bijection with strands they induce in the uncountable tree.

A path is a sequence of nodes where each node is followed by one of its successors. We can define an equivalent concept for real junctions.

Definition. A **real junction sequence (RJS)** is a sequence of real junctions where each real junction in the sequence is followed by one of its nearest real descendants

Definition. A **rooted real junction sequence** is a real junction sequence starting with the real junction root

Lemma 3.5.3. *There is a bijection between the rooted real junction sequences of an uncountable binary tree and the strands of a complete binary tree*

Proof. We know from lemma 3.5.2 that there is a bijective map φ between the real junctions **RJ** of an uncountable binary tree and the nodes of a complete binary tree **CBT** that preserves descent. That is when $\varphi(r_0) = c_0$, where r_0 is the real junction root in RJ and c_0 is the root node of RJT, $r_i \succ r_j \Leftrightarrow \varphi(r_i) \succ \varphi(r_j)$ where $r_i, r_j \in RJ$

A strand of a complete binary tree is defined by a denumerable sequence of successor nodes starting from the root. So a strand C in CBT consists of nodes $c_0, c_1, c_2, ...c_n, ...$ where c_0 is the root node and $c_i \succ c_j$ when $i \leq j$

A rooted real junction sequence R in an uncountable tree consists of real junctions $r_0, r_1, r_2, ...r_n, ...$ where r_0 is the real junction root, each $r_i \in RJ$ and $r_i \succ r_j$ when $i \leq j$

Applying the bijective map φ to the nodes of R we get a strand C in CBT $\varphi(r_0), \varphi(r_1), \varphi(r_2), ...\varphi(r_n), ...$ where $r_i \succ r_j \Leftrightarrow \varphi(r_i) \succ \varphi(r_j)$, that is φ preserves the order of descent. Also since φ is bijective $\varphi(\varphi^{-1}(r_i)) = r_i$. So applying φ^{-1} to C gives back the original strand R.

Let RRJS be the set of rooted real junction sequences and STR be the set of strands in CBT. Define $\rho : RRJS \mapsto STR$ by mapping the junctions in a rooted real junction sequence from RRJS to nodes in CBT using φ. Then $\rho(R) = C$ produces a strand $C \in STR$ which is isomorphic to R and $\rho(\rho^{-1}(R)) = R$.

It follows that ρ is a bijection. □

3.5.3 Rooted Real Junction Sequences ⇔ Real Strands

A real junction sequence contains only real junctions and their descendant real junctions. If we extend it to include the other nodes in the tree that are ancestors of those real junctions, it will induce a strand.

Definition. A **real strand** of an uncountable binary tree is a real junction sequence together with the nodes that are ancestors of the real junctions in the sequence

Lemma 3.5.4. *In a tree there is a unique path between any two nodes*

Proof omitted: For example refer Rosen, *Discrete Mathematics and Its Applications*, p746

Lemma 3.5.5. *The unique path between a real junction r of a real junction sequence and the root node of the tree includes all of r's ancestors*

Proof. Let r_j be a real junction in a real strand. Let r_i be a node which is an ancestor of r_j. (r_i may be a real junction or just a normal node.) By lemma 3.5.4 there is a unique path from r_j to

r_i. There is also a unique path from r_i to r_0, the root node. It follows that the unique path from r_j to r_0 includes r_i.

Therefore the unique path between any real junction r of a real strand and the root node includes all of r's ancestors. □

Lemma 3.5.6. *A real strand is a strand*

Proof. From lemma 3.5.5 the unique path between any real junction r_k of a real strand and the root node includes all of r_k's ancestors. The nodes in that path also define a path in the opposite direction from the root node back to r_k. There is also a path from the root to r_k's immediate descendant r_{k+1} which includes the path to r_k since r_k is r_{k+1}'s ancestor.

Since a real strand contains an infinite sequence of real junctions each of which has the previous real junction as an ancestor, it follows that a real strand defines an infinite path from the root node and so is also a strand. □

A real junction sequence does not necessarily contain all the real junctions that are ancestors of the real junctions in the sequence. So two different real junction sequences can induce the same real strand. However a rooted real junction sequence does contain all of the ancestors of the real junctions in the sequence, so two different rooted real junction sequences will always define different strands.

Lemma 3.5.7. *A rooted real junction sequence includes all the real junctions that are ancestors of the other real junctions in the sequence*

Proof. From lemma 3.5.5 the unique path from any real junction r of an RJS R to the root node of the tree includes all of r's ancestors including r_0 the real junction root. If R is additionally a rooted RJS then r_0 will be included in R. Any other real junctions which are r's ancestors must be descended from r_0. Since R is an RJS starting from r_0 which includes r and since r's ancestors are on the unique path from r_0 to r they will also be included in R . □

Lemma 3.5.8. *There is a bijection between the rooted real junction sequences of an uncountable binary tree and its real strands*

Proof. If two RJS R_1 and R_2 are different, one of them say R_2 must contain a real junction r not in R_1. If R_1 is a rooted RJS then r cannot be an ancestor of the other real junctions in R_1 since by lemma 3.5.7 a rooted RJS contains all its ancestors. In that case R_1 and R_2 must induce different real strands since the real strand that R_2 induces will contain r but the one induced by R_1 will not.

Let $\rho : RRJS \mapsto SRS$ where RRJS is the set of rooted RJS in an uncountable tree and SRS is the set of real strands. We have shown that $R_1 \neq R_2 \Rightarrow \rho(R_1) \neq \rho(R_2)$. So ρ is injective. Since every RJS induces a real strand, ρ is also surjective.

Therefore ρ is bijective. □

3.5.4 Real Strands $\Leftrightarrow$ Strands of a Complete Binary Tree

Lemma 3.5.9. *The strands of an uncountable binary tree are at least equinumerous with the strands of a complete binary tree*

Proof. Lemma 3.5.3 shows that the rooted real junction sequences of UNC have a bijection with the strands of CBT. Lemma 3.5.8 shows that they have a bijection with UNC's real strands. Therefore there is a bijection between the real strands of UNC and the strands of CBT. Lemma 3.5.6 shows that UNC's real strands are a subset of its strands.

Therefore the strands of UNC are at least equinumerous with the strands of CBT. □

Lemma 3.5.10. *The strands of a complete binary tree are at least equinumerous with* $\mathbb{R}$

Proof. Every node of CBT has two successors which we can label 0 and 1. If the root node is labelled 0 then the complete set of paths through CBT will each start at 0 and together they then include every possible sequence of nodes labelled 0 and 1. As such there is a 1:1 correspondence between the infinite paths through CBT and the strings in SS. By lemma 3.2.2 there is a bijective map between (0,1) and a subset of SS.

Since (0,1) has cardinality $\mathfrak{c}$, the infinite paths of CBT are at least equinumerous with $\mathbb{R}$. □

3.6 Conclusion

Theorem 2. *An uncountable set of real numbers is equinumerous with* $\mathbb{R}$

Proof. From lemma 3.3.1, if a subset of $\mathbb{R}$ is uncountable it has a bijection with the strands of an uncountable binary tree.

From lemma 3.5.9 the strands of an uncountable binary tree are at least equinumerous with those of a complete binary tree, which by lemma 3.5.10 are at least equinumerous with $\mathbb{R}$.

Therefore since a subset of $\mathbb{R}$ is at most equinumerous with $\mathbb{R}$, an uncountable set of real numbers is equinumerous with $\mathbb{R}$. □

References

Freiwald, Ronald C. *An Introduction to Set Theory and Topology*. 2014.
Rosen, Kenneth H. *Discrete Mathematics and Its Applications*. 7th Ed. McGraw-Hill, 2012.
Stillwell, John. *The Real Numbers*. Springer, 2013.