A Universe with large Continuum, global Σ-Uniformization and a projective Well-Order of its Reals

Stefan Hoffelner[1]*

[1]TU Wien
June 16, 2026

Abstract

We construct a model in which the continuum has size κ for a regular cardinal κ and in which the Σ^1_n-uniformization property holds simultaneously for every $n \geqslant 2$. Additionally this model has a Δ^1_3-definable well-order of its reals.

1 Introduction

This article deals with two notions of the real numbers which are ruled out if we assume large cardinals or the axiom of projective determinacy PD. The first notion is the projectively definable well-order of the real numbers, and the second one is global Σ-uniformization. Given $A \subset \omega^\omega \times \omega^\omega$, we say that f is a uniformization (or a uniformizing function) of A if f is a partial function such that

$$dom(f) = pr_1(A) = \{x \in 2^\omega \,:\, \exists y((x,y) \in A\}$$

and the graph of f is a subset of A.

Definition 1.1 (Uniformization Property)**.** *We say that the projective pointclass $\Gamma \in \{\Sigma^1_n \mid n \in \omega\} \cup \{\Pi^1_n \mid n \in \omega\}$ has the uniformization property iff every Γ-set in the plane admits a uniformization whose graph is in Γ, i.e. the relation $(x,y) \in f$ is in Γ.*

Due to Moschovakis (see [15] 39.9) PD implies that Π^1_{2n+1} and Σ^1_{2n+2}-sets have the uniformization property. By the Martin-Steel theorem (see [16], Theorem 13.6.), the assumption of infinitely many Woodin cardinals outright implies PD, and hence large cardinals fully settle the behaviour of

*This research was funded in whole by the Austrian Science Fund (FWF) Grant-DOI 10.55776/P37228.

the uniformization property within the projective hierarchy. The size of the continuum can be arbitrarily changed, modulo the few constraints imposed by ZFC, without altering the zig-zag pattern of the uniformization property within the projective hierarchy.

Global Σ-uniformization is the statement that for all $n \geqslant 2$, Σ^1_n-sets have the uniformization property. It is well-known that L satisfies global Σ-uniformization, which follows from the fact that the canonical well-order of the constructible reals is not only Δ^1_2-definable but also a *good* well-order. Recall that a Δ^1_n-definable well-order $<$ of the reals is good if $<$ has ordertype ω_1 and the relation $<_I \subset (\omega^\omega)^2$ defined via

$$x <_I y \Leftrightarrow \{(x)_n : n \in \omega\} = \{z : z < y\}$$

where $(x)_n$ is some fixed recursive partition of x into ω-many reals, is a Δ^1_n-definable relation. Addison (see [1]) observed that Δ^1_n-definable well-orders imply Σ^1_m-uniformization for every $m \geqslant n$.

For a long time, good projective well-orders were the only tool to derive global Σ-uniformization, hence the question of whether global Σ-uniformization is consistent with the continuum being arbitrarily large remained open. Goal of this work is to prove that

Theorem 1.2. *Assume $V = L$ and let κ be a regular cardinal. Then there is a generic extension of L where $2^{\aleph_0} = \kappa$ and the Σ^1_n-uniformization property holds for every $n \geqslant 2$ simultaneously.*

Adding an additional layer of coding forcings one can also obtain the next theorem which is the one we will prove in this article:

Theorem 1.3. *Assume $V = L$ and let κ be a regular cardinal. Then there is a generic extension of L where $2^{\aleph_0} = \kappa$ and the Σ^1_n-uniformization property holds for every $n \geqslant 2$ simultaneously and the reals have a Δ^1_3-definable well-order.*

Universes with some interesting properties while having a projectively definable well-order of the reals are a very active topic in current set theory (see e.g. [3], [4] or [6] for a very small subset of the existing literature) and presumably started with L. Harrington's work in [5].

This article employs the technique pioneered in [10] which uses infinitary logic and a copy mechanism to produce a method which forces global Σ-uniformization in the absence of a a good projective well-order. This technique is combined with the coding machinery from [8] and [11], which a bit surprisingly, can also be used to find universes with a large continuum while keeping a robust coding mechanism. There are naturally some differences though, the method in [10] uses a countable support iteration. It is well-known that countable support iterations only allow a continuum of size

$\leqslant \aleph_2$, thus we are compelled to work with finite support iterations in order to obtain a universe with a large continuum.

We end with a short description of how the article is organized. Section 2 introduces the building blocks of our coding machine. It shows the existence of an ω_1-length, independent sequence of Suslin trees which is definable in L, a result which must have been known before but we could not find a proof so we included the one from [9]. Section 3 defines coding forcings which can be used to switch a Σ^1_3-predicate from false to true. These coding forcings are as in [8] and [11]. Building on these codings we define a family of Σ^1_n-predicates for every $n \geqslant 3$ in section 3 which is inspired by [10]. These predicates can be controlled in a very precise way using iterations of the coding forcings from section 3, and will be used to produce the uniformizing functions we need when forcing the global Σ-uniformization. In section 5 we finally define the iteration which will produce the desired universe which shows our main theorem. That the iteration does what it should do is shown in section 6.

2 Independent Suslin trees in L, almost disjoint coding

The coding method of our choice utilizes Suslin trees, which can be generically destroyed in an independent way of each other.

Definition 2.1. *Let $\vec{T} = (T_\alpha : \alpha < \kappa)$ be a sequence of Suslin trees. We say that the sequence is an independent family of Suslin trees if for every finite set of pairwise distinct indices $e = \{e_0, e_1, ..., e_n\} \subset \kappa$ the product $T_{e_0} \times T_{e_1} \times \cdots \times T_{e_n}$ is a Suslin tree again.*

Note that an independent sequence of Suslin trees $\vec{T} = (T_\alpha : \alpha < \kappa)$ has the property that if $A \subset \kappa$ and we form $\prod_{i \in A} T_i$ with finite support, where each T_i denotes the forcing we obtain if we force with the nodes of the tree as conditions using the tree order as the partial order, then in the resulting generic extension $V[G]$, for every $\alpha \notin A$, $V[G] \models$ "T_α is a Suslin tree". To see this, we assume the opposite, namely there is an $\alpha \notin A$ such that $V[G]$ thinks that T_α is not Suslin anymore. We note that $V[G]$ is a generic extension of V obtained with a forcing with the ccc (see [7], Lemma 51 for a more general argument of this type). So there is an $\aleph_1$-sized antichain of T_α in $V[G]$. But then the finitely supported product of the form $(\prod_{i \in A} T_i) \times T_\alpha$ can not have the ccc, which is a contradiction to the assumed independence of $\vec{T}$.

One can easily force the existence of independent sequences of Suslin trees with products of Jech's or Tennenbaum's forcing, or with just products of ordinary Cohen forcing. On the other hand independent sequences of length ω_1 already exist in L.

Theorem 2.2. *Assume that $\aleph_1 = \aleph_1^L$ and that $(M, \in)$ is a transitive, uncountable model of ZFC^- + "$\aleph_1$ exists". Then there is an independent sequence $\vec{T} = (T_\alpha \mid \alpha < \omega_1)$ of L-Suslin trees and the sequence $\vec{T}$ is uniformly $\Sigma_1(\{\omega_1\})$-definable over M. To be more precise, there is a Σ_1-formula ϕ with ω_1 as the unique parameter, which does not depend on the model M, such that the relation $\{(t, \gamma, \eta) \mid \gamma, \eta < \omega_1 \wedge t \in T_\eta^\gamma\}$, where T_η^γ denotes the γ-th level of $T_\eta \in \vec{T}$, is definable over M using ϕ.*

Proof. We first argue in L and later argue that our to be presented construction relativizes to any M as in the proposition. We fix the canonical $\diamondsuit$-sequence $(a_\alpha \subset \alpha \mid \alpha < \omega_1)$ in L. Next we alter the usual construction of a Suslin tree from $\diamondsuit$ to construct an ω_1-sequence of Suslin trees $\vec{T} = (T_\alpha \mid \alpha < \omega_1)$.

We fix a Δ_1-definable bijection of the set of pairs of countable ordinals with ω_1. We use this bijection in the usual way to code sets of pairs of countable ordinals with sets of countable ordinals. Under this identification, trees will be coded as subsets of ordinals.

We assume that $\alpha < \omega_1$ is a limit ordinal, and split the set $a_\alpha \subset \alpha$ into two parts via considering the even and odd ordinals of a_α. We assume first that if we recursively split again the even ordinals of a_α into three sets $A, B, C \subset \text{Even}(a_\alpha)$, then A codes a wellorder of ordertype $\beta < \alpha < \omega_1$, B codes a wellorder of ordertype $\gamma < \alpha$ and C codes a finite subset of ω_1. This indicates that we aim to define the β-th level of T_γ as follows.

We assume first that β is a limit ordinal and T_γ^i and T_η^i for $\eta \in C$ has been defined already for each $i < \beta$. Now we consider the odd ordinals of a_α and assume that $\text{Odd}(a_\alpha)$ codes a set $A \subset \bigcup_{i<\beta} T_\gamma^i \times \prod_{\eta \in C} T_\eta^i$ which is an antichain there.

We define the β-th level T_γ^β of T_γ such that it seals off the antichain A. To be more specific we chose $T_\gamma^\beta \times \prod_{\eta \in C} T_\eta^\beta$ in such a way that A remains a maximal antichain in all further extensions of $T_\gamma^\beta \times \prod_{\eta \in C} T_\eta^\beta$.

Otherwise we just extend T_γ^β via adding top nodes on countably many branches through T_γ^β.

If β is not a limit ordinal we define T_γ^β under the assumption that $T_\gamma^{\beta-1}$ is already defined and let T_γ^β be just the result of putting ω-many new nodes above each node of $T_\gamma^{\beta-1}$.

We let $T_\gamma := \bigcup_{\beta < \omega_1} T_\gamma^\beta$ and claim that $(T_\gamma \mid \gamma < \omega_1)$ is an independent sequence of Suslin trees in L.

Indeed, if $A \in L$ is an antichain in some $\prod_{\gamma \in e} T_\gamma$, then there is a constructible club of ordinals α such that $A \cap \alpha = a_\alpha$ and $\text{Even}(a_\alpha)$ codes A, B, C where A codes β, B codes γ and C codes $e \backslash \{gamma\}$ and $\text{Odd}(a_\alpha)$ codes a maximal antichain in $\bigcup_{i<\beta} T_\gamma^i \times \prod_{\eta \in C} T_\eta^i$. But then $\text{Odd}(a_\alpha)$ got sealed off and $\prod_{\gamma \in e} T_\gamma$ has no uncountable antichains, so is a Suslin tree in L.

The definability of $\vec{S}$ comes from the fact that the canonical $\diamondsuit$-sequence in L is $\Sigma_1(\{\omega_1\})$-definable. We can use L_{ω_1} (which is $\Sigma_1(\{\omega_1\})$ to correctly define $\diamondsuit$ over it and consequentially $\vec{S}$ becomes definable over L_{ω_1} as well. The above considerations can be simulated correctly already in any transitive, uncountable M which models ZFC^-, as it will compute L_{ω_1} correctly and the rest of the construction is performed inside the latter model. $\square$

We briefly introduce the almost disjoint coding forcing due to R. Jensen and R. Solovay. We will identify subsets of ω with their characteristic function and will use the word reals for elements of 2^ω and subsets of ω respectively. Let $D = \{d_\alpha : \alpha < \aleph_1\}$ be a family of almost disjoint subsets of ω, i.e. a family such that if $r, s \in D$ then $r \cap s$ is finite. Let $X \subset \omega$ be a set of ordinals. Then there is a ccc forcing, the almost disjoint coding $\mathbb{A}_D(X)$ which adds a new real x which codes X relative to the family D in the following way

$$\alpha \in X \text{ if and only if } x \cap d_\alpha \text{ is finite.}$$

Definition 2.3. *The almost disjoint coding $\mathbb{A}_D(X)$ relative to an almost disjoint family D consists of conditions $(r, R) \in [\omega]^{<\omega} \times D^{<\omega}$ and $(s, S) < (r, R)$ holds if and only if*

1. *$r \subset s$ and $R \subset S$.*
2. *If $\alpha \in X$ and $d_\alpha \in R$ then $r \cap d_\alpha = s \cap d_\alpha$.*

We shall briefly discuss the L-definable, $\aleph_1^L$-sized almost disjoint family of reals D we will use throughout this article. The family D is the canonical almost disjoint family one obtains when recursively adding the $<_L$-least real x_β not yet chosen and replace it with $d_\beta \subset \omega$ where this d_β is the real which codes the initial segments of x_β using some recursive bijections between ω and $\omega^{<\omega}$. The definition of D is uniform over any uncountable, transitive ZF^--models M with, as we can correctly compute L up to $\aleph_1^L$ inside M and then apply the above definition inside L's version of M. Even more is true, if M is a countable, transitive model of ZF^- + " $\aleph_1$ exists and $\aleph_1 = \aleph_1^L$", then M will compute $D \upharpoonright \omega_1^M$ in a correct way. The reason is again, that M can define an initial segment of L correctly which suffices to calculate $D \upharpoonright \omega_1^M$.

Last we state a short lemma which will be helpful when showing that our coding forcings work the way they should.

Lemma 2.4. *Let T be a Suslin tree and let $\mathbb{A}_D(X)$ be the almost disjoint coding which codes a subset X of ω_1 into a real with the help of an almost disjoint family of reals D of size $\aleph_1$. Then*

$$\mathbb{A}_D(X) \Vdash T \text{ is Suslin}$$

holds.

Proof. This is clear as $\mathbb{A}_D(X)$ has the Knaster property, thus the product $\mathbb{A}_D(X) \times T$ is ccc and T must be Suslin in $V[\mathbb{A}_D(X)]$. $\square$

3 Coding forcings

We continue with the construction of the appropriate notions of forcing which we want to use in our proof. The goal is to first define a coding forcings Code(x) for reals x, which will force for x that a certain Σ^1_3-formula $\Phi(x)$ becomes true in the resulting generic extension. The coding method is basically the same as in [8], [13] and [9], with the only difference that we add more $\mathbb{C}(\omega_1)$-subsets to create room for the continuum to eventually become κ.

We fix our regular cardinal κ. In a first step, we add κ-many ω_1-Cohen subsets with a countably supported product,

$$\mathbb{P}^1 := \prod_{\alpha<\kappa} \mathbb{C}(\omega_1).$$

Note that this forcing is itself σ-closed so no reals are added and $\vec{S}$ is still an independent sequence of Suslin trees. In a second step, we force over $L[\mathbb{P}^1]$ to destroy all members of $\vec{S}$ via generically adding an ω_1-branch, that is, we form $\mathbb{P}^0 := \prod_{\alpha\in\omega_1} S_\alpha$ with finite support. Note that this is an $\aleph_1$-sized, ccc forcing over L and also $L[\mathbb{P}^1]$, and the forcing is independent of the actual model in which it is computed. The two step iteration can be thus conceived as a product of two factors $(\prod_{i<\omega_1} \mathbb{C}(\omega_1))^L \times \prod_{\alpha\in\omega_1} S_\alpha$. In the generic extension $\aleph_1$ is preserved and CH remains to be true.

We use W to denote this generic extension of L, that is

$$W = L[\mathbb{P}^0 \times \mathbb{P}^1].$$

Let $x \in W$ be a real and let $\eta < \omega_1$. The forcing Code(x, η) which codes the real (x into $\vec{S}$ is defined as the almost disjoint coding forcing of a specific set $Y \subset \omega_1$, that is

$$\text{Code}(x, \eta) := \mathbb{A}(Y).$$

We will define the crucial set $Y \subset \omega_1$ now.

To ease notation we let $g \subset \omega_1$ be g_η for $\eta < \kappa$, where g_η is the η-th coordinate of the $\prod_{\alpha<\kappa} \mathbb{C}(\omega_1)$-generic filter over $L[\mathbb{P}^0]$. We let $\rho : ([\omega_1]^\omega)^L \to \omega_1$ be some canonically definable, constructible bijection between these two sets. We use ρ and g to define the set $h \subset \omega_1$, which eventually shall be the set of indices of ω-blocks of $\vec{S}$, where we "code up the characteristic function of the real x", the latter slogan will be made precise in a moment. Let

$$h := \{\rho(g \cap \alpha) \,:\, \alpha < \omega_1\}$$

and let

$$\begin{aligned} A := &\{\omega\gamma + 2n \mid \gamma \in h, n \notin x\} \cup \\ &\{\omega\gamma + 2n + 1 \mid \gamma \in h, n \in x\}. \end{aligned}$$

Let $X \subset \omega_1$ be chosen such that it codes the following objects:

- The set $A \subset \omega_1$.
- Some set $\{b_\beta \subset S_\beta \mid \beta \in A\}$ of ω_1-branches. We demand that for every $\beta \in A$, b_β is a $L[\mathbb{P}^0]$-generic branch for the forcing $S_\beta \in \vec{S}^1$.

If $L_\eta[X]$ is some uncountable ZFC^--model, and if we assume that $\gamma \in h$ then $L_\eta[X]$ can read off x via looking at the ω-block of $\vec{S}$-trees starting at γ and determine which tree has an ω_1-branch in $L_\eta[X]$. Thus we say that x is coded into $\vec{S}$ at the ω-block starting at γ in such a situation. To be more specific $L_\eta[X]$ satisfies the following formula:

$(*)_{\gamma,x}$ $n \in x$ if and only if $S_{\omega\cdot\gamma+2n+1}$ has an ω_1-branch, and $n \notin x$ if and only if $S_{\omega\cdot\gamma+2n}$ has an ω_1-branch.

Indeed if $n \notin x$ then we added a cofinal branch through $S_{\omega\cdot\gamma+2n}$. If on the other hand $S_{\omega\cdot\gamma+2n}$ does not have an ω_1-branch in $L_\eta[X]$ then we must have added an ω_1-branch through $S_{\omega\cdot\gamma+2n+1}$ as we always add an ω_1-branch through either $S_{\omega\cdot\gamma+2n+1}$ or $S_{\omega\cdot\gamma+2n}$ and adding branches through some S_α's will not affect that some S_β remain Suslin in $L[X]$, as $\vec{S}$ is independent.

We note that we can apply an argument resembling David's trick [1] in this situation. We rewrite the information of $X \subset \omega_1$ as a subset $Y \subset \omega_1$ using the following line of reasoning. It is clear that any transitive, $\aleph_1$-sized ZFC^- model N of the form $N = L_\eta[X]$ will be able to first define $\vec{S}$ correctly and also correctly decode out of X all the information regarding x being coded at each ω-block of $\vec{S}$ starting at every $\gamma \in h$. Consequently, if we code the model $(N, \in)$ which contains X, as a set $X_N \subset \omega_1$, then for any uncountable β such that $L_\beta[X_N] \models \mathsf{ZFC}^-$:

$$L_\beta[X_N] \models \text{"The model decoded out of } X_N \text{ satisfies } (*)_{\gamma,x} \text{ for every } \gamma \in h\text{"}.$$

In particular there will be an $\aleph_1$-sized ordinal β as above and we can fix a club $C \subset \omega_1$ and a sequence $(M_\alpha : \alpha \in C)$ of countable elementary submodels of $L_\beta[X_N]$ such that

$$\forall \alpha \in C (M_\alpha \prec L_\beta[X_N] \wedge M_\alpha \cap \omega_1 = \alpha)$$

Now let the set $Y \subset \omega_1$ code the pair (C, X_N) such that the odd entries of Y should code X_N and if $E(Y)$ denotes the set of even entries of Y and $\{c_\alpha : \alpha < \omega_1\}$ is the enumeration of C then

1. $E(Y) \cap \omega$ codes a well-ordering of type c_0.
2. $E(Y) \cap [\omega, c_0) = \varnothing$.

[1] see [2] for the original argument, where the strings in Jensen's coding machinery are altered such that certain unwanted universes are destroyed. This destruction is emulated in our context as seen below.

3. For all β, $E(Y) \cap [c_\beta, c_\beta + \omega)$ codes a well-ordering of type $c_{\beta+1}$.

4. For all β, $E(Y) \cap [c_\beta + \omega, c_{\beta+1}) = \emptyset$.

The following formula $\sigma(v_0)$ is now true for our real x:

$\sigma(x) \equiv$ For any countable transitive model M of "ZF^- and $\aleph_1$ exists" such that $\omega_1^M = (\omega_1^L)^M$ and $Y \cap \omega_1^M \in M$, M can construct its version of the universe $L[Y \cap \omega_1^M]$, and the latter will see that there is an $\aleph_1^M$-sized transitive model $N \in L[Y \cap \omega_1^M]$ which models $(*)_{\gamma,x}$ for $\aleph_1^M$-many $\gamma \in M$.

Thus we have a local version of the property $(*)_{\gamma,x}$.

We have finally defined the desired set Y and now we define

$$\text{Code}(x, \eta) := \mathbb{A}(Y)$$

relative to our previously defined, almost disjoint family of reals $D \in L$ (see the paragraph after Definition 2.5) to code the set Y into one real r. This forcing only depends on the subset of ω_1 we code, thus $\mathbb{A}_D(Y)$ will be independent of the surrounding universe in which we define it, as long as it has the right ω_1 and contains the set Y.

The effect of the coding forcing $\text{Code}(x, \eta)$ is that it generically adds a real r such that

$\Psi(r, x) \equiv$ For any countable, transitive model M of "ZFC^- and $\aleph_1$ exists", such that $\omega_1^M = (\omega_1^L)^M$ and $r \in M$, M can construct its version of $L[r]$, denoted by $L[r]^M$, which in turn thinks that there is a transitive ZFC^--model N of size $\aleph_1^M$ such that N believes $(*)_{\gamma,x}$ for an $\aleph_1^M$-sized set of ordinals γ.

Indeed, if r and M are as above, then M and $L[r]^M$ will compute the almost disjoint family D up to the real indexed with $\omega_1 \cap M$ correctly, as discussed below the definition 2.3. As a consequence, $L[r]^M$ will contain the set $Y \cap \omega_1^M$, where $Y \subset \omega_1$ is as in $\sigma(x)$. So in $L[Y \cap \omega_1^M]$, there is an $\aleph_1^M$-sized, transitive N which models $(*)_{\gamma,x}$ for $\aleph_1^M$-many $\gamma \in M$, as claimed.

Note that $\Psi(r, x)$ is a Π^1_2-formula in the parameters r and x, as the set $h \cap M \subset \omega_1^M$ is coded into r. We say in the above situation that r witnesses that the real x *is written into* $\vec{S}$, or that x *is coded into* $\vec{S}$. To summarize our discussion, given an arbitrary real x, then our forcing $\text{Code}(x, \eta)$, when applied over W, will add a real r which will turn the Π^1_2-formula $\Psi(r, x)$ for r, x into a true statement in $W^{\text{Code}(x,\eta)}$.

The projective and local statement $\Psi(r, x)$, if true, will determine how certain inner models of the surrounding universe will look like with respect to branches through $\vec{S}$. That is to say, if we assume that $\Psi(r, x)$ holds then r also witnesses that the assertion of $\Psi(r, x)$ remains true even if we replace the countable, transitive models M in the statement of $\Psi(r,x)$ with any transitive model M of the theory "ZFC^-+ $\aleph_1$ exists and $\aleph_1 = \aleph_1^L$". In other words we can drop the assumption on the countability of M in $\Psi(r,x)$, provided $\Psi(r,x)$ is true. Indeed if we assume that there would be an uncountable, transitive M which models ZFC^- and "$\aleph_1$ exists" and $\omega_1 = \omega_1^L$, which additionally contains r and which witnesses that the conclusion of $\Psi(r,x)$ is false. Then by Löwenheim-Skolem, there would be a countable $N \prec M$, $r \in N$ which we can transitively collapse to obtain the transitive $\bar{N}$. But $\bar{N}$ would witness that the conclusion of $\Psi(r,x)$ is not true for every countable, transitive model, which is a contradiction to our assumption that $\Psi(r,x)$ is true.

Consequently, the real r carries enough information that the universe $L[r]$ will see that certain trees from $\vec{S}$ have branches in that $L[r]$ will decode, with the help of the almost disjoint family $D \in L$ a set $h \subset \omega_1$ such that

$$n \in x \Rightarrow L[r] \models \forall \gamma \in h\,(\text{"}S_{\omega\gamma+2n+1} \text{ has an } \omega_1\text{-branch"}).$$

and

$$n \notin x \Rightarrow L[r] \models \forall \gamma \in h\,(\text{"}S_{\omega\gamma+2n} \text{ has an } \omega_1\text{-branch"}).$$

Indeed, the universe $L[r]$ will see that there is a transitive model N of "ZF^-+ $\aleph_1$ exists and $\aleph_1 = \aleph_1^L$" which believes $(*)_{\gamma,x}$ for every $\gamma \in h \subset \omega_1$, the latter being coded into r. But by upwards Σ_1-absoluteness, and the fact that N can compute $\vec{S}$ correctly, if N thinks that some tree in $\vec{S}$ has a branch, then $L[r]$ must think so as well.

The definition of the forcing $\text{Code}(x,\eta)$ exhibits a significant degree of absoluteness. Specifically, its definition is entirely independent of the universe in which it is computed, provided that universe contains the necessary set $Y \subset \omega_1$ (which encodes the relevant branches tied to x, the club C, etc., as defined above). We will leverage this property shortly.

Our strategy involves iteratively applying these coding forcings. The intention is to encode progressively more reals into the structure $\vec{S}$, thereby populating our target Σ^1_3-set, which comprises all reals coded into $\vec{S}$. It is essential that this iterative process does not inadvertently code reals that were not explicitly intended. The following result confirms this is the case.

We let $\Phi(x)$ denote the Σ^1_3-statement "x is coded into $\vec{S}$" that is

$$\Phi(x) := \exists r \Psi(r,x).$$

Via spelling out $\Psi(r,x)$ we see that $\Phi(x)$ is the following formula:

$$\begin{aligned}\Phi(x) \equiv \exists r(\forall M(&M \text{ countable, transitive } M \models \mathsf{ZFC}^- \wedge \omega_1^M = (\omega_1^L)^M \rightarrow \\ &M \models \text{"}L[r] \models r \text{ codes with the help of the a.d. familiy } D \restriction \omega_1^M \\ &\text{a transitive, } \aleph_1^M\text{-sized } \mathsf{ZFC}^-\text{-model } N' \text{ such that } N' \text{ satisfies} \\ &(*)_{\gamma,x} \text{ for } \aleph_1^M\text{-many ordinals } \gamma\text{"}))\end{aligned}$$

Consider a finite support iteration $(\mathbb{P}_\beta, \dot{\mathbb{Q}}_\beta \mid \beta < \kappa)$ where each $\dot{\mathbb{Q}}_\beta$ is forced by $\mathbb{P}_\beta$ to be a coding forcing $\mathrm{Code}(\dot{x}_\beta, \dot{\eta}_\beta)$ for some $\mathbb{P}_\beta$-name for a real $\dot{x}_\beta$ and a sequence of names of ordinals $\dot{\eta}_\beta$ which should be forced by each condition to be injective. Let G be $\mathbb{P}_\kappa$-generic over $W = L[g^0][g^1]$. Within the generic extension $L[g^0][g^1][G]$, we define the set of reals *intentionally coded* by the iteration $\mathbb{P}_\kappa$ and generic G as $\{\dot{x}_\beta^G \mid \beta < \kappa\}$.

Lemma 3.1. *Let $\mathbb{P} \in L[g^0 * g^1]$ be a finite support iteration of coding forcings* $\mathrm{Code}(\dot{x}_\beta, \dot{\eta}_\beta)$ *of length κ such that $\Vdash \forall \alpha \neq \beta(\dot{\eta}_\alpha \neq \dot{\eta}_\beta)$. Let $G \subset \mathbb{P}$ be generic over $L[g^0 * g^1]$, and let $\{x_\beta \mid \beta < \kappa\}$ be the set of reals intentionally coded by $\mathbb{P}^G$. Then, in the model $W[G]$, the set of reals x satisfying $\Phi(x)$ is precisely $\{x_\beta \mid \beta < \kappa\}$.*

Proof. We work within $L[g^0][g^1][G]$. Assume, seeking a contradiction, that there exists a real x such that $\Phi(x)$ holds, but x is not among the intentionally coded reals, i.e., $x \notin \{x_\beta \mid \beta < \kappa\}$. The truth of $\Phi(x)$ means there is a real r witnessing it. Since $\mathbb{P}$ is a finite support iteration, any real in the extension $W[G]$ is actually present in an extension generated by a countable sub-iteration. Therefore, there exists a countable set $I \subset \kappa$ such that both x and r are elements of the intermediate model $W[G_I]$, where $G_I = G \cap \mathbb{P}_I$ and $\mathbb{P}_I$ is the sub-iteration of $\mathbb{P}$ restricted to indices in I.

Furthermore, coding each x_i (for $i \in I$) involves countably many elements from g^0 and the according elements (which are uncountably many with uncountable complement) of g^1. Thus, x and r actually reside in a model of the form $L[g^0 \upharpoonright K][g^1 \upharpoonright J][G_I]$ for some countable $K \subset \omega_1$ and some $J \subset \omega_1$ of size $\aleph_1$ whose complement also has size $\aleph_1$. Let $W' = L[g^0 \upharpoonright K][g^1 \upharpoonright J][G_I]$. We conduct the remainder of the argument within this ZFC universe W', constructed inside $L[g^0][g^1][G]$.

Since r witnesses $\Phi(x)$ in W', according to our discussion above, r encodes information implying the existence of an unbounded set $h \subset \omega_1$ such that for all $\gamma \in h$:

$$n \in x \implies L[r] \models \text{“}S_{\omega\gamma+2n+1} \text{ has an } \omega_1\text{-branch.”}$$
$$n \notin x \implies L[r] \models \text{“}S_{\omega\gamma+2n} \text{ has an } \omega_1\text{-branch.”}$$

Because x is distinct from every x_β with $\beta \in I$, and as the set of starting points of ω-blocks of $\vec{S}$ where the x_β's are coded forms an almost disjoint family on ω_1 (i.e. two distinct elements have countable intersection) by construction (they correspond to the almost disjoint family we forced using the $\prod_{i<\kappa} \mathbb{C}(\omega_1)$ and the bijection $\rho[\omega_1]^\omega \to \omega_1$), there must exist an ordinal α (in fact there are $\aleph_1$-many such α's but we don't need that) and some $n \in \omega$ where

$$L[r] \models \text{“}S_{\omega\alpha+2n+1} \text{ has an } \omega_1\text{-branch,”}$$

but for every $\beta \in I$, if r_β is the real witnessing $\Phi(x_\beta)$, then:

$$L[r_\beta] \models \text{“}S_{\omega\alpha+2n+1} \text{ does not have an } \omega_1\text{-branch.”}$$

We fix such an α and claim that no real R exists within the model $W' = L[g^0 \upharpoonright K][g^1 \upharpoonright J][G_I]$ such that $L[R] \models$ "$S_{\omega\alpha+2n+1}$ has an ω_1-branch." This will yield the desired contradiction.

The claim holds because the forcing $(\mathbb{P}^0 \upharpoonright K) * (\mathbb{P}^1 \upharpoonright J) * (\mathbb{P} \upharpoonright I)$ and its generic filter $(g^0 \upharpoonright K) * (g^1 \upharpoonright J) * (G_I)$ used to form W' from L did not explicitly add a branch through $S_{\omega\alpha+2n+1}$ as we chose α such that $\omega\alpha + 2n + 1 \notin J$. Consequently all the forcings used in $(g^0 \upharpoonright K) * (g^1 \upharpoonright J) * (G \upharpoonright I)$, i.e. forcing with Suslin tree from $\vec{T}$ which are not the tree $T_{\omega\alpha+2n+1}$, almost disjoint coding forcing and $\mathbb{C}(\omega_1)^L$ preserve the Suslin tree $T_{\omega\alpha+2n+1}$, so this tree will remain Suslin in W'. But if there would be a real R such that $L[R] \models$ "$S_{\omega\alpha+2n+1}$ has an ω_1-branch", then by upwards absoluteness there would be such an ω_1-branch also in W' which is a contradiction.

So no real $x \neq x_\beta$ could satisfy $\Phi(v_0)$ as asserted. □

As a final observation relevant for later use: the argument in the preceding proof holds even if we intersperse Cohen forcing $\mathbb{C}$ (or any other Suslin tree preserving, ccc forcing notion) within the finite support iteration. That is, the lemma remains valid for iterations whose factors are either coding forcings or standard Cohen forcing. This flexibility will be useful later, particularly for achieving goals related to increasing the size of the continuum.

4 Suitable Σ^1_n-predicates

We shall use the Σ^1_3-predicate "being coded into $\vec{S}$" (we will often write just "being coded" for the latter) to form suitable Σ^1_n-predicates Φ^n for every $n \in \omega$. These predicates share the following properties:

1. $L \models \forall x \neg(\Phi^n(x))$
2. For every real $x \in L$, there is an iteration of coding forcings $\text{Code}^n(x) \in L$ such that after forcing with it, $L^{\text{Code}^n(x)} \models \Phi^n(x)$, and for every real $y \neq x$, $L^{\text{Code}^n(x)} \models \neg\Phi^n(y)$.

Most importantly, these properties remain true even when iterating the (iterations of coding forcings $\text{Code}^n(x_i)$ for a sequence of (names of) reals.

The predicates $\Phi^n(x)$ will be defined now. In the following we let (x, y) denote a real z which recursively codes the pair of reals consisting of x and y. Likewise $(x_0, ..., x_n)$ is defined.

- $\Phi^3(x, y, m) \equiv \exists a_0((x, y, m, a_0)$ is coded into $\vec{S})$.
- $\Phi^4(x, y, m) \equiv \exists a_0 \forall a_1((x, y, m, a_0, a_1)$ is not coded into $\vec{S})$.
- $\Phi^5(x, y, m) \equiv \exists a_0 \forall a_1 \exists a_2((x, y, m, a_0, a_1, a_2)$ is coded into $\vec{S})$.
- $\Phi^6(x, y, m) \equiv \exists a_0 \forall a_1 \exists a_2 \forall a_3((x, y, m, a_0, a_1, a_2, a_3)$ is not coded into $\vec{S})$.

- ...
- ...
- $\Phi^{2n}(x, y, m) \equiv \exists a_0 \forall a_1 ... \forall a_{2n-3}((x, y, m, a_0, ..., a_{2n-3})$ is not coded into $\vec{S}$).
- $\Phi^{2n+1}(x, y, m) \equiv \exists a_0 \forall a_1 ... \exists a_{2n-2}((x, y, m, a_0, ..., a_{2n-2})$ is coded into $\vec{S}$).
- ...
- ...

Each predicate Φ^n is exactly Σ^1_n. In the choice of our Σ^1_n-formulas $\Phi^n(x)$, we encounter again a periodicity phenomenon, that is two different cases depending on $n \in \omega$ being even or odd, a theme which is pervasive in this area. It is clear that for each predicate Φ^n and each given real x there is a way to create a universe in which $\Phi^n(x)$ becomes true using our coding forcings. We just need to iterate the relevant coding forcings using countable support. For $n = 3$ we just need one coding forcing, for $n \geqslant 3$ our iteration will have inaccessible length in order to catch our tail. As shown already, our coding method allows us to exactly code the tuple of reals we want to code, without accidentally adding some unwanted information. Thus the next lemma is straightforward to prove, so we just state it.

Lemma 4.1. *Let $n \in \omega$ and let x be a real in W. Then there is a forcing $\mathrm{Code}^n(x)$ which is an iteration of our coding forcings $\mathrm{Code}(x_i, \eta_i)$ such that if $G \subset \mathrm{Code}^n(x)$ is generic, $W[G]$ will satisfy $\Phi^n(x)$ and for every $y \neq x$, $W[G] \models \neg\Phi^n(y)$. This property can be iterated, that is it remains true if we replace W with $W[G]$ in the above.*

5 Defining the desired forcing iteration

We shall work towards finding the right iteration which will eventually prove the main theorem. The technique to force the global Σ-uniformzation property is a sort of a copying mechanism where we use the coding forcings to code up infinite conjunctions of projective formulas. The construction is pioneered in [10] and applied there in a different context using a different coding technique. It is flexible enough to be applicable in our context with our coding machinery as well.

5.1 Forcing global $\Sigma-$uniformization

We will need the following notions. For an arbitrary real x, list the triples $(x, \dot{y}^0, \dot{a}^0_0), (x, \dot{y}^1, \dot{a}^1_0), (x, \dot{y}^2, \dot{a}^2_0), ...)$ according to our fixed well-order $<$. To be more precise we list all the $\mathbb{P}$-names of reals which are recursive codes for triples of $\mathbb{P}$-names of reals, where $\mathbb{P}$ is a κ-sized partial order. The list should have the property that names of longer iterations always appear after the names of the shorter iterations. We also assume that for every x, $\dot{a}_0 = \dot{y}_0 = 0$ to rule out a degenerate case in the following. This has technical advantages as will become clear as we proceed in the argument. The upshot of this is that whenever $(x, a_0 = 0)$ is in our Σ^1_n-set A_m and the membership is already witnessed by $a_0 = 0$, then we let 0 be the x-value of our uniformizing function of A_m. Thus we can, without loss of generality, always work under the Σ^1_{n-1}-assumption that 0 does not witness that $(x, 0)$ is an element of A_m. As we aim for a Σ^1_n-definition of the uniformizing function, this assumption is harmless with regards of complexity.

For each ordinal $\alpha < 2^{\aleph_0}$ we fix bijections $\pi_\alpha : (2^{\aleph_0})^\alpha \to 2^{\aleph_0}$ (we assume w.l.o.g that such a bijection exist as we always force its existence if we blow up the contiuum with, say, Cohen forcing, to size $|\alpha|$ and then use a maximal almost disjoint family $\mathcal{F}$ and almost disjoint coding forcing relative to $\mathcal{F}$ to ensure that in the resulting model $|(2^{\aleph_0})^\alpha| = 2^{\aleph_0}$). These bijections are of course sensitive to the surrounding universe and we assume that, as we iteratively enlarge our universe via a forcing, the bijections extend each other so that they cohere. To be more precise if $\pi^\beta_\alpha : (2^{\aleph_0})^\alpha \to 2^{\aleph_0}$ is our chosen family of bijections[2] in the universe $V[G_\beta]$ which arises at the β-th stage of our iteration, and $\pi^\gamma_\alpha : (2^{\aleph_0})^\alpha \to 2^{\aleph_0}$, $\beta < \gamma$ is the family of bijections we fix at the γ-th stage $V[G_\gamma]$, then for every $\alpha < (2^{\aleph_0})^{V[G_\beta]}$,

$$\pi^\gamma_\alpha \upharpoonright (2^\omega \cap V[G_\beta]) = \pi^\beta_\alpha.$$

Let

$$F : \kappa \to \kappa^\omega$$

be some bookkeeping function in $L[g^0 * g^1]$ which shall be our ground model and which guides our iteration. The choice of F does not really matter, it is sufficient to assume that every $x \in \kappa^\omega$ has an unbounded pre-image under F. We assume that we have defined already the following list of notions:

- We defined already our iteration $\mathbb{P}_\beta \in L[g^0 * g^1]$ up to stage β.
- We picked a $\mathbb{P}_\beta$-generic filter G_β for $\mathbb{P}_\beta$ and work, as usual, over $L[g^0 * g^1][G_\beta]$.
- In $L[g^0 * g^1][G_\beta]$ we picked a family $\{\pi_\alpha \mid \alpha < 2^{\aleph_0}\}$ of bijections of $(2^{\aleph_0})^\alpha$ and $2^{\aleph_0}$. These bijections should cohere with the older families of bijections, as discussed above. We assume without loss of generality

[2] These families of bijections are responsible for the fact that the coding construction presented in this paper and the coding construction which forces the, say, Π^1_3-uniformization property or the Π^1_3-reduction property (see [11] and [8]) can not be combined.

that such a family exists, if not then we alter $L[g^0 * g^1][G_\beta]$ with a ccc forcing (namely e.g. an iteration of Cohen forcing followed by an iteration of almost disjoint coding forcing relative to some maximal, almost disjoint family of reals) such that such a family exists in the new universe.

We assume that the bookkeeping function F at β hands us an ω tuple $(\gamma_0, \gamma_1, ...)$ of ordinals below κ which corresponds to an ω-tuple $(\dot{x}, \dot{m}, \dot{\alpha}, \dot{b}_0, \dot{b}_1, \dot{b}_2, ...)$ where $\dot{x}$ and $\dot{b}_n$ are $\mathbb{P}_\beta$-names of a reals and $\dot{m}$ and $\dot{\alpha}$ are $\mathbb{P}_\beta$-names of a natural number and of an ordinal bigger than 0 and less than the size of the continuum respectively (in fact at each stage we will only need finitely many of those names of reals $\dot{b}_i$ and the rest of the information will be discarded). The correspondence is given by demanding that $\gamma_0 \leqslant \beta$ and $\dot{x}$ is the γ_1-th name of a real in $L[g^0][g^1][G_{\gamma_0}]$, $\dot{b}_0$ is the γ_2-th name of a real in $L[g^0][g^1][G_{\gamma_0}]$ and so forth.

We let $x = \dot{x}^{G_\beta}$, and define b_n, m, α and the G_β-evaluation of our list of names of reals $(x, \dot{y}_0, \dot{a}_0), ...$ accordingly. The natural number m is the Gödelnumber $\#\varphi$ of a Σ^1_n-formula $\varphi(x, y)$ in two free variables. Our goal is to define the forcing $\dot{\mathbb{Q}}_\beta$ we want to use at stage β, and to define the notion of $\alpha_\beta + 1$-based forcing. We consider various cases for our definition.

5.1.1 Case 1, odd projective level

In the first case we write $\varphi_m = \exists a_0 \forall a_1 ... \exists a_{2n-2} \psi(x, y, a_0, ..., a_{2n-2})$, where ψ is a Π^1_2-formula (and φ_m is a Σ^1_{2n+1} formula for $n \geqslant 1$) for the formula with Gödelnumber m which is handed to us by the value $F(\beta)$. We repeat our assumption form above that $F(\beta)$ determines a tuple $b_1, ...b_{2n-2}$ of real numbers, x, and an ordinal $0 < \alpha < 2^{\aleph_0}$ with the assigned reals a^α_0 and y^α which stem from our wellordered list of triples $(x, y^0, a^0_0), (x, y^1, a^1_0), ...(x, y^\alpha, a^\alpha_0), ...$ [3] a we fixed in advance.

Recall that we have our fixed bijection $\pi_\alpha : (2^{\aleph_0})^\alpha \rightarrow 2^{\aleph_0}$.

Letting $\pi^{-1}_\alpha(b_k) := (b^\eta_k \mid \eta < \alpha)$ then case 1 is the condition that

$$\neg\psi(x, y^\eta, a^\eta_0, b^\eta_1, b^\eta_2, ..., b^\eta_{2n-2}) \text{ is true for every } \eta < \alpha.$$

Then we use the coding forcing

$$\text{Code}(\#\psi, x, y^\alpha, a^\alpha_0, b_1, b_2, ..., b_{2n-2}) =: \dot{\mathbb{Q}}_\beta$$

as the β-th factor of our iteration.

[3]This list should be injective, that is whenever a triple $(x, (\dot{y}_\alpha)^{G_\beta}, (\dot{a}_\alpha)^{G_\beta})$ appears which appeared already earlier in our list we drop it.

5.1.2 Case 2, odd projective level

We again let $\varphi_m = \exists a_0 \forall a_1 ... \exists a_{2n-2} \psi(x, y, a_0, ..., a_{2n-2})$ and we assume that the bookkeeping F at β hands us a tuple $b_1, ...b_{2n-2}$ of real numbers, x, and $0 < \alpha < 2^{\aleph_0}$ with the assigned reals a_0^α and y^α.

Case 2 is the condition that if $\pi_\alpha^{-1}(b_k) := (b_k^\eta \mid \eta < \alpha)$ then

$$\psi(x, y^\eta, a_0^\eta, b_1^\eta, b_2^\eta, ..., b_{2n-2}^\eta) \text{ is true for an } \eta < \alpha.$$

In this situation we force with the trivial forcing at the β-th stage.

5.1.3 Case 3, even projective level

This is the dual case to the first one, but this time m is the Gödelnumber of a formula which belongs to an even projective level. We write

$$\varphi_m \equiv \exists a_0 \forall a_1 ... \forall a_{2n-3} (\psi(x, y, a_0, ..., a_{2n-3})),$$

where ψ is a Σ^1_2-formula and φ_m is a Σ^1_{2n}-formula.

The bookkeeping F at β hands us a tuple $b_1, ..., b_{2n-3}$ of real numbers, x, and $0 < \alpha < 2^{\aleph_0}$ with the assigned reals a_0^α and y^α.

Recall that we have our fixed bijection $\pi_\alpha : (2^{\aleph_0})^\alpha \to 2^{\aleph_0}$.

Case 3 is the condition that if $\pi_\alpha^{-1}(b_k) := (b_k^\eta \mid \eta < \alpha)$ then

$$\neg\psi(x, y^\eta, a_0^\eta, b_1^\eta, b_2^\eta, ..., b_{2n-3}^\eta) \text{ is true for every } \eta < \alpha.$$

We do not use a forcing in this case.

5.1.4 Case 4, even projective level

This is dual to case 2.

We assume that if $\pi_\alpha^{-1}(b_k) := (b_k^\eta \mid \eta < \alpha)$ then

$$\psi(x, y^\eta, a_0^\eta, b_1^\eta, b_2^\eta, ..., b_{2n-3}^\eta) \text{ is true for an } \eta < \alpha.$$

In this situation we use the coding forcing of the form

$$\text{Code}(\#\varphi_m, x, y^\alpha, a_0^\alpha, b_1, ..., b_{2n-3})$$

as the β-th forcing $\dot{\mathbb{Q}}_\beta$ in our iteration.

We use finite support for this iteration and consequently the iteration is a ccc forcing over $L[g^0 * g^1]$ hence preserves cofinalities.

This ends the definition of our iteration. What is left is to show that the definition works in that it will produce a universe where σ_{even} and σ_{odd} serve as definitions of uniformizing functions, when applied in the right context.

5.2 Forcing the continuum being large

Forcing a large continuum will be achieved by the standard method. As in the last section, we assume that we are at stage $\beta < \kappa$ of our iteration and we have defined already $\mathbb{P}_\beta$. Let G_β be a $\mathbb{P}_\beta$ generic filter over $L[g^0 * g^1]$ and our fixed bookkeeping function at stage β hands us some default set, say 0. In this situation we will just force with plain Cohen forcing,

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \mathbb{C}$$

and move on to the next step of the iteration.

5.3 Towards a Δ^1_3-well-order of the reals

Adding forcings which will eventually produce a Δ^1_3-definable well-order of the reals of the final universe does not cause problems as well. We assume that $\beta < \kappa$, $G_\beta \subset \mathbb{P}_\beta$ a generic filter over $L[g^0][g^1]$ and the bookkeeping $F(\beta)$ yields a pair of names of ordinals $(\dot{\eta}_0, \dot{\eta}_1)$ which evaluate with the help of G_β to $\eta_0, \eta_1 < \kappa$ and we assume that $\eta_0 \leqslant \beta$. We assume that in $L[g^0][g^1][G_\eta]$, the η_1-th (in L's canonical global well-order) pair of $\mathbb{P}_{\eta_0}$-names of reals is $(\dot{b}_0, \dot{b}_1)$. We proceed as follows. First we evaluate $\dot{b}_0^{G_\beta} = b_0$ and $\dot{b}_1^{G_\beta} = b_1$ and consider the $<$-least (again, $<$ denotes L's canonical global well-order) forcing names σ_0, σ_1 such that $\sigma_0^{G_\beta} = b_0$ and $\sigma_1^{G_\beta} = b_1$. If $\sigma_0 < \sigma_1$ then we use

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(b_0, b_1),$$

and if $\sigma_1 < \sigma_0$ we let

$$\dot{\mathbb{Q}}_\beta^{G_\beta} := \text{Code}(b_1, b_0).$$

6 Properties of the resulting universe

We shall prove the main properties of our just defined universe now. Let G_κ be the final generic filter for our just defined, κ-length iteration $(\mathbb{P}_\beta, \dot{\mathbb{Q}}_\eta \mid \beta \leqslant \kappa, \eta < \kappa)$ with finite support. Note that each factor $\dot{\mathbb{Q}}_\beta$ of our iteration is either an almost disjoint coding forcing of the form Code(x), for some real x, or Cohen forcing. Thus $\mathbb{P}_\kappa$ has the ccc and all cardinals are preserved. As we will force κ-many times with Cohen forcing, $\mathbb{P}_\kappa$ will force that the continuum is $\geqslant \kappa$.

We now turn to prove that the global Σ-uniformization property is forced by $\mathbb{P}_\kappa$.

Lemma 6.1. *Let G_κ denote a generic filter for the full, κ-length iteration.*

1. *Then $W[G_\kappa]$ satisfies that whenever φ_m is on an odd projective level, say $\varphi_m = \exists a_0 \forall a_1 ... \exists a_{2n-2} \psi(x, y, a_0, ..., a_{2n-2})$ and $(x, y^\alpha, a_0^\alpha)$ is such that*

$$W[G_\kappa] \models \forall a_1 \exists a_2 ... \exists a_{2n-2} \psi(x, y^\alpha, a_0^\alpha)$$

Then for each $\beta > \alpha$

$$W[G_\kappa] \models \forall a_1 ... \exists a_{2n-2}((\#\psi, x, y^\beta, a_0^\beta, a_1, .., a_{2n-2}) \text{ is not coded })$$

2. *If* $\varphi_m = \exists a_0 \forall a_1 ... \forall a_{2n-3} \psi(x, y, a_0, ..., a_{2n-3})$ *is a formula on the even projective level and* $(x, y^\alpha, a_0^\alpha)$ *is such that*

$$W[G_\kappa] \models \varphi_m(x, y^\alpha, a_0^\alpha)$$

Then for each $\beta > \alpha$

$$W[G_\kappa] \models \forall a_1 \exists a_2 ... \forall a_{2n-3}((\#\psi, x, y^\beta, a_0^\beta, a_1, .., a_{2n-3}) \text{ is coded })$$

Proof. We will only proof the first item, as the second is shown in the dual way. Let $\beta > \alpha$. We know that

$$W[G_\kappa] \models \forall a_1 \exists a_2 ... \exists a_{2n-2} \psi(x, y^\alpha, a_0^\alpha)$$

holds true. The way we defined our iteration, in particular case 2, implies that we can use the bijection $\pi_\beta : (2^\omega)^\beta \to 2^\omega$, to translate between true infinite disjunctions of the form

$$\bigvee_{\eta < \beta} \psi(x, y^\eta, a_0^\eta, b_1^\eta, ..., b_{2n-2}^\eta)$$

and

"$(\#\psi, x, y^\beta, a_0^\beta, b_1, b_2, ..., b_{2n-2})$ is not coded into $\vec{S}$",

where $b_i = \pi_\beta((b_i^\eta \mid \eta < \beta))$. As $\forall a_1 \exists a_2 ... \exists a_{2n-2} \psi(x, y^\alpha, a_0^\alpha)$ is true, this witnesses that our assertion $\forall a_1 \exists a_2 ... \exists a_{2n-2}$ $((\#\psi, x, y^\beta, a_0^\beta, a_1, ..., a_{2n-2})$ is not coded into $\vec{S})$ is true in $W[G_\kappa]$.

□

Lemma 6.2. *Again* G_κ *denotes a generic filter for the entire,* κ*-length iteration.*

1. *Assume that* $\varphi(v_0, v_1)$ *is in* Σ^1_{2n+1}*. Let* $\varphi = \exists a_0 \forall a_1 ... \exists a_{2n-2} \psi(v_0, v_1)$*, where* $\psi \in \Pi^1_2$*. If* α *is least such that*

$$W[G_\kappa] \models \forall a_1 ... \exists a_{2n-2} \psi(x, y^\alpha, a_0^\alpha, ..., a_{2n-2}),$$

then

$$W[G_\kappa] \models \exists a_1 \forall a_2 ... \forall a_{2n-2}((\#\psi, x, y^\alpha, a_0^\alpha, a_1, a_2, ..., a_{2n-2}) \text{ is coded.})$$

2. *Assume that* $\varphi(v_0, v_1) \in \Sigma^1_{2n}$ *and* $\varphi(v_0, v_1) = \exists a_0 ... \forall a_{2n-3} \psi(v_0, v_1)$ *for* $\psi \in \Sigma^1_2$*. If* α *is least such that*

$$W[G_\kappa] \models \forall a_1 ... \exists a_{2n-3} \psi(x, y^\alpha, a_0^\alpha, ..., a_{2n-3}),$$

then

$$W[G_\kappa] \models \exists a_1 ... \forall a_{2n-3}((\#\psi, x, y^\alpha, a_0^\alpha, a_1, ..., a_{2n-3}) \text{ is not coded.})$$

Proof. We shall show only the first item of the lemma, as the second one is proved in the dual way. As α is the first ordinal for which

$$W[G_\kappa] \models \forall a_1 ... \exists a_{2n-2} \psi(x, y^\alpha, a_0^\alpha, ..., a_{2n-2}),$$

we know that $\forall \beta < \alpha$, $W[G_\kappa] \models \exists a_1 \forall a_2, ..., \forall a_{2n-2}(\neg \psi(x, y^\beta, a_0^\beta, a_1, ...))$. In particular, for every $\beta < \alpha$ there is a real a_1^β such that for every a_2^β there is a real a_3^β and so on such that $\neg\psi$ holds. Using the bijection π_α, we can find an $a_1 = \pi_\alpha((a_1^\beta)_{\beta<\alpha})$ which has the property that for any real a_2 there will be a real $a_3 = \pi_\alpha(a_3^\beta)_{\beta<\alpha})$ such that for any real a_4 and so on $\neg\psi$ is true.

But this translates via the case 1 rule of our iteration to the assertion that

$$W[G_\kappa] \models \exists a_1 \forall a_2 ... \forall a_{2n-2}((\# \psi, x, y^\alpha, a_0^\alpha, a_1, a_2, ..., a_{2n-2}) \text{ is coded.})$$

This is what we wanted. □

Lemma 6.3. *In $W[G_\kappa]$ the Σ^1_{n+1}-uniformization property holds true for every $n \geqslant 1$.*

Proof. Again we will only consider the case for the odd projective levels. Let $\varphi \equiv \exists a_0 \forall a_1 ... \exists a_{2n-2}(\psi(x, y, a_0, a_1, ..., a_{2n-2}))$ be an arbitrary Σ^1_{2n+1}-formula in two free variables where ψ is Π^1_2. Let x be a real such that there is a real y with $L[G_\kappa] \models \varphi(x, y)$. We list all the triples $(x, y^\alpha, a_0^\alpha)$ according to our well-order $<$. Remember that we defined y^0, a_0^0 to be both 0. If a_0^0 witnesses that $\forall a_1 \exists a_2 ... (\psi(x, 0, 0, a_1, ...)$ is true, which is a Π^1_{2n}-formula, then $\varphi(x, 0)$ will be true in $W[G_\kappa]$ and is the value of our uniformizing function there.

If $a_0^0 = 0$ does not witness that $\forall a_1 \exists a_2 ... (\psi(x, 0, 0, a_1, a_2, ...)$ is true, which is a Σ^1_{2n}-formula then let $\alpha > 0$ be least such that

$$W[G_\kappa] \models \forall a_1 ... (\psi(x, y^\alpha, a_0^\alpha, a_1, ...)).$$

Then by the last Lemma

$$W[G_\kappa] \models \exists a_1 \forall a_2 ... \forall a_{2n-2}((\# \psi, x, y^\alpha, a_0^\alpha, ...) \text{ is coded})$$

holds true. Note that this formula is Σ^1_{2n+1}.

Yet, by the penultimate Lemma, for each $\beta > \alpha$

$$W[G_\kappa] \models \forall a_1 \exists a_2 ... \exists a_{2n-2}((\# \psi, x, y^\beta, a_0^\beta, a_1, ..) \text{ is not coded}).$$

So (x, y^α) is the unique pair satisfying the Σ^1_{2n+1}-formula

$$\exists a_0((\forall a_1 \exists a_2 ... \exists a_{2n-2}(\psi(x, y, a_0, ...) \wedge \\ \neg(\forall a_1 \exists a_2 ... \exists a_{2n-2}((\# \psi, x, y, a_0, a_1, ... a_{2n-2} \text{ is not coded}))$$

Indeed, if $\beta > \alpha$, then the triple (x, y^β, a_0^β) can not satisfy the second subformula above. And if $\beta < \alpha$, then (x, y^β, a_0^β) will not satisfy

$$\forall a_1 \exists a_2 ... \exists a_{2n-2} \psi(x, y^\beta, a_0^\beta, a_1, ..., a_{2n-2})$$

as α was chosen to be least.

So to summarize the Σ^1_{2n+1}-formula

$$\forall a_1 \exists a_2 ... (\psi(x, 0, 0, a_1, a_2, ...) \vee$$
$$(\exists a_1 \forall a_2 ... (\neg \psi(x, 0, 0, a_1, a_2, ...) \wedge \exists a_0 ((\forall a_1 \exists a_2 ... \exists a_{2n-2} (\psi(x, y, a_0, ...) \wedge$$
$$\neg(\forall a_1 \exists a_2 ... \exists a_{2n-2} ((\# \psi, x, y, a_0, a_1, ... a_{2n-2} \text{ is not coded})))$$

defines the uniformizing function for φ □

The next lemma is an immediate consequence of the fact that we define our well-order of the reals using the global L-well-order. In particular we will never be in a situation where at one stage β_0 of our iteration, we have $\dot{b}_0^{G_{\beta_0}} < \dot{b}_1^{G_{\beta_0}}$ and at a later stage $\beta_1 > \beta_0$ we have $\dot{b}_1^{G_{\beta_0}} > \dot{b}_0^{G_{\beta_0}}$. Thus, given two arbitrary reals b_0 and b_1 in $L[g^0][g^1][G_\kappa]$, either the pair (b_0, b_1) got coded into $\vec{S}$ or the pair (b_1, b_0) got coded into $\vec{S}$.

Lemma 6.4. *In $W[G_\kappa]$ the reals have a Δ^1_3-definable well-order via*

$$b_0 < b_1 \text{ iff } (b_0, b_1) \text{ is coded into } \vec{S}$$

and

$$b_1 < b_0 \text{ iff } (b_1, b_0) \text{ is coded into } \vec{S}.$$

7 Open problems

The proof of the theorem can not be combined with forcing axioms. Indeed already trying to force MA while applying the proof of of the theorem will cause problems which can not be repaired, as MA implies that for all subsets of ω_1 there are reals which are almost disjoint codes for them. This ruins our coding in that we lose control over which reals we want to code and which not.

Question 1. *Let κ be an arbitrary regular cardinal. Is there a model of* MA*, $2^{\aleph_0} = \kappa$ and global Σ-uniformization?*

The coding methods should be flexible enough to also force a global failure of Σ-uniformization.

Question 2. *Using the coding methods from this work or relatives. Can one force a universe where Σ^1_n-uniformization fails for every $n \geqslant 2$? Is it possible to additionally force* MA*?*

References

[1] J. W. Addison *Some Consequences of the Axiom of Constructibility.* Fundamenta Mathematicae 46, pp. 337-357, 1959.

[2] R. David *A very absolute Π^1_2-real singleton.* Annals of Mathematical Logic 23, pp. 101-120, 1982.

[3] V. Fischer, S. D. Friedman and L. Zdomskyy *Projective wellorders and MAD families with large continuum.* Annals of Pure and Applied Logic 162, pp. 853-862, 2011.

[4] V. Fischer, S. D. Friedman and Y. Khomskii *Measure, category and projective wellorder.* Journal of Logic and Analysis 6)pp. 1-25, 2014.

[5] L. Harrington *Long projective wellorderings.* Annals of Mathematical Logic, 12, pp. 1–24, 1977.

[6] S. D. Friedman and S. Hoffelner *A Σ^1_4-Well-Order of the Reals with NS_{ω_1} saturated.* Journal of Symbolic Logic 84, No. 4, pp. 1466-1483, 2019.

[7] S. Hoffelner *NS_{ω_1} Δ_1-definable and saturated.* Journal of Symbolic Logic 86 (1), pp. 25 - 59, 2021.

[8] S. Hoffelner *Forcing the Π^1_3-Reduction Property and a Failure of Π^1_3-Uniformization*, Annals of Pure and Applied Logic 174 (8), 2023.

[9] S. Hoffelner *A Failure of Π^1_{n+3}-Reduction in the Presence of Σ^1_{n+3}-Separation*, Submitted.

[10] S. Hoffelner *The global Σ^1_{n+2}-Uniformization Property and* BPFA Advances in Mathematics, 2025

[11] S. Hoffelner *Forcing the Π^1_n-Uniformization Property*, submitted.

[12] S. Hoffelner *Forcing Axioms and the Uniformization Property* Annals of Pure and Applied Logic 175 (10), 2024.

[13] S. Hoffelner *Forcing the Σ^1_3-Separation Property.* Journal of Mathematical Logic, 2022.

[14] T. Jech *Set Theory. Third Millenium Edition.* Springer 2003.

[15] A. Kechris *Classical Descriptive Set Theory.* Springer 1995.

[16] R. D. Schindler *Set Theory. Exploring Independence and Truth* Springer 2014.