Forcing with Random Variables and Proof Complexity (Reading Group, June 13 & June 20)

Yuval Filmus

June 17, 2013

1 One-sorted models

In the previous week, we went over the construction of the models $K(F)$. Here we give a concrete example: the model $K(F_{\text{rud}})$ whose two-sorted version makes an appearance later on in the book.

Our starting point is some ω_1-saturated non-standard model of arithmetic $\mathcal{M}$ which has names for all *natural functions*, which are functions $\mathbf{N}^k \to \mathbf{N}$ for some integer k. We take a non-standard integer n. We call a member of $\mathcal{M}$ *subexponential* if it is at most 2^{n^ϵ} for some infinitesimal ϵ. The set of all subexponentials is denoted $\mathcal{M}_n$ (since it's downward-closed, it forms a cut). A natural function has *subexponential growth* if given subexponential inputs, its output is subexponential. (Each natural function extends to a function in $\mathcal{M}$ which satisfies the same first-order properties.) For example, polynomials have subexponential growth. The set of all functions having subexponential growth is denoted L_n.

We take an integer n and let Ω_{rud} consist of all definable subsets of $[n]$. A *decision tree* is a binary tree whose inner nodes are labeled $i \overset{?}{\in} \omega$ (where $i \in [n]$), its edges are labels YES and NO, and its leaves are labeled with subexponentials. A decision tree is *shallow* if its height is n^ϵ for some infinitesimal ϵ. The set F_{rud} consists of all functions computed by shallow decision trees. Note that all these functions are coded by elements of $\mathcal{M}$.

The set F satisfies a crucial property: it is L_n-closed. This means that if $f(x_1, \ldots, x_k)$ has subexponential growth and $\beta_1, \ldots, \beta_k \in F_{\text{rud}}$ then the function $f(\beta_1, \ldots, \beta_k)$, defined by $f(\beta_1, \ldots, \beta_k)(\omega) = f(\beta_1(\omega), \ldots, \beta_k(\omega))$, is also in F_{rud}. To see this, we simply compose the trees.

2 Two-sorted models

In the sequel, we use two-sorted models which have an addition string sequence, which more properly consists of $\mathcal{M}$-finite strings. This will be useful for formulating the pigeonhole principle, as well as crucial for expressing quantifier elimination, which is the technique we will use to prove induction on our models.

We will denote number sort variables using lowercase letters and string sort variables using uppercase letters. We can think of string variables in two ways: either as sequences of numbers, or as sets of integers. For a number term t and a string variable X, $X(t)$ is the value of X at t, where X is considered as a sequence. We will also use the shorthand $t \notin X$ for $X(t) = 0$. We call X a *set* if X is $\{0,1\}$-valued. If X is a set then $t \in X$ iff $X(t) = 1$.

The strings that we consider will usually be bounded: for some t, $X(x) \neq 0 \to x < t$. In this case we write $X < t$ and say that X is a string of length t. (A string of length t is also a string of length s for all $s > t$.) The related notations $\forall X < t$ and $\exists X < t$ are shorthands for $\forall X (X < t \to \cdots)$ and $\exists X (X < t \wedge \cdots)$, respectively.

In the model $K(F, G)$, F represents the number sort and G represents the string sort. The construction of G from F is always the same, and we illustrate it by extending the model $K(F_{\text{rud}})$ to $K(F_{\text{rud}}, G_{\text{rud}})$.

For each subexponential m and m-tuple $\beta_0, \dots, \beta_{m-1} \in F_{\text{rud}}$ (which belongs to $\mathcal{M}$), we define an element $\Theta = \Theta^{\beta_0,\dots,\beta_{m-1}}$ of G by

$$\Theta(\alpha) = \lambda\omega. \begin{cases} \beta_{\alpha(\omega)}(\omega) & \text{if } \alpha(\omega) < m, \\ 0 & \text{otherwise.} \end{cases}$$

Here $\alpha \in F_{\text{rud}}$, and the notation $\lambda\omega.f(\omega)$ signifies a function mapping $\omega \in \Omega_{\text{rud}}$ to $f(\omega) \in \mathcal{M}_n$. The set G_{rud} consists of all $\Theta^{\beta_0,\dots,\beta_{m-1}}$ for all sequences $\langle \beta_0, \dots, \beta_{m-1} \rangle$ in $\mathcal{M}$ whose length is subexponential.

In order to understand the definition, consider first the case of constant α. For a subexponential x, if $x < m$ then $\Theta(x) = \beta_x$, and otherwise $\Theta(x) \equiv 0$. Thus Θ behaves like a string of length m. When α is not constant, $\alpha(\omega)$ codes the position of the string being accessed.

There is another way of looking at this definition, by switching the order of arguments. For $\omega \in \Omega_{\text{rud}}$ define a function $\Theta_\omega : [m] \to \mathcal{M}_n$ by

$$\Theta_\omega = \langle \beta_0(\omega), \dots, \beta_{m-1}(\omega) \rangle.$$

In other words, Θ is a random sequence of length m, each entry of which has a uniform definition, in the sense that it belongs to F_{rud}. Under this definition we have

$$\Theta(\alpha) = \lambda\omega.\Theta_\omega[\alpha(\omega)].$$

Here the square brackets signify the i'th element in a sequence (in this case $i = \alpha(\omega)$), and we use the convention that elements which are out of range are simply zero (we can imagine that the sequences are encoded using the Gödel encoding $2^{x_0}3^{x_1}5^{x_2}\cdots$). In the future we will use parentheses instead of square brackets whenever we are sure that $\alpha(\omega)$ is in range.

The crucial property satisfied by this definition is that $\Theta(\alpha) \in F_{\text{rud}}$. This can be seen by taking the decision tree for α and replacing the leaves by either 0 or decision trees for the appropriate β_i. (To make this argument formal, we have to notice that the trees in the m-tuple $\beta_0, \dots, \beta_{m-1}$ have a maximal height, and so there is a uniform subpolynomial bound on their height.) Another important property is that G_{rud} contains all constant strings of subexponential length, analogously to F_{rud} containing all constant subexponential numbers.

3 Defining truth

We now extend the definition of truth from $K(F)$ to $K(F, G)$. We will use the following notation: for a predicate $P(\omega)$, $\langle\!\langle P(\omega) \rangle\!\rangle = \{\omega \in \Omega : P(\omega)\}$; I is the ideal of infinitesimally small sets in the Boolean algebra $\mathcal{B}$. The definitions for $K(F)$ were:

- $[\![\alpha = \beta]\!] = \langle\!\langle \alpha(\omega) = \beta(\omega) \rangle\!\rangle / I$.
- For a relation R, $[\![R(\alpha_1, \dots, \alpha_k)]\!] = \langle\!\langle R(\alpha_1(\omega), \dots, \alpha_k(\omega)) \rangle\!\rangle / I$.
- $[\![\neg t]\!] = \neg[\![t]\!]$, $[\![s \wedge t]\!] = [\![s]\!] \wedge [\![t]\!]$, $[\![s \vee t]\!] = [\![s]\!] \vee [\![t]\!]$.
- $[\![\forall x A(x)]\!] = \bigwedge_{\alpha \in F} [\![A(\alpha)]\!]$.
- $[\![\exists x A(x)]\!] = \bigvee_{\alpha \in F} [\![A(\alpha)]\!]$.

The latter two exist since the Boolean algebra is complete.

The extensions to $K(F, G)$ are:

- $[\![\Theta = \Xi]\!] = \langle\!\langle \Theta_\omega = \Xi_\omega \rangle\!\rangle / I$.
- $[\![\forall X A(X)]\!] = \bigwedge_{\Theta \in G} [\![A(\Theta)]\!]$.
- $[\![\exists X A(X)]\!] = \bigvee_{\Theta \in G} [\![A(\Theta)]\!]$.

4 Lower bounds using forcing

How do we prove lower bounds using this framework? For simplicity, we only consider the AC^0 case. Our goal is to prove lower bounds for a Frege system in which all lines have constant depth, i.e. AC^0-Frege. To this end, we consider the corresponding two-sorted uniform proof theory V^0. This theory is axiomatized by the usual Peano axioms (without induction), and the following two axiom schemes:

Comprehension: For any bounded formula A, $\exists X \forall y < x (y \in X \leftrightarrow A(y, z_1, \dots, z_k))$. (Here $x, z_1, \dots, z_k$ are parameters.)

Induction: For any bounded formula B, $[B(0, y_1, \dots, y_k) \wedge \forall x (B(x, y_1, \dots, y_k) \rightarrow B(x+1, y_1, \dots, y_k))] \rightarrow \forall x B(x, y_1, \dots, y_k)$. (Here $y_1, \dots, y_k$ are parameters.)

Bounded formulas are only allowed to have bounded number quantifiers $\forall x < t$ and $\exists x < t$, and *no* string quantifiers.

Proofs in V^0 correspond to subexponential size constant depth AC^0-Frege proof, a phenomenon known as *propositional translation*. However, we will be interested in a slightly different property of V^0: it can prove the reflection principle for AC^0-Frege. To explain the reflection principle, we have to describe some predicates, depending on a parameter $d \in \mathbf{N}$:

- $F_d(y, Y, z)$ is a bounded formula coding "Y is a depth-d formula of length y on z variables".
- $P_d(x, X, y, Y, z)$ is a bounded formula coding "X is a depth-d AC^0-Frege proof of length x of formula Y of length y on z variables".
- $S_d(y, Y, z, Z)$ is a bounded formula coding "Z is a satisfying assignment for the depth-d formula Y of length y on z variables".

The reflection principle states that $\forall Z (F_d(y, Y, z) \wedge P_d(x, X, y, Y, z) \rightarrow S_d(y, Y, z, Z))$. For each $d \in \mathbf{N}$, V^0 can prove this principle.

The pigeonhole principle can be described by a bounded formula $\mathrm{PHP}(x, R)$ whose meaning is "R doesn't encode an injection from $[x+1]$ to $[x]$" (a weaker principle states that R doesn't encode a bijection from $[x+1]$ onto $[x]$). It can also be encoded by a propositional formula $\mathrm{PHP}_x(r_{0,0}, \dots, r_{x,x-1})$ on $(x+1)x$ variables encoding the graph of R. The theory V^0 proves that for all R, $S_d(|\mathrm{PHP}_x|, \mathrm{PHP}_x, (x+1)x, R) \leftrightarrow \mathrm{PHP}(x, R)$.

Suppose that for some $d \in \mathbf{N}$ and all large enough $m \in \mathbf{N}$ there exist depth-d AC^0-Frege proofs Π_m of PHP_m of size at most $s(m)$, where $s(m) = 2^{m^{o(1)}}$ (that is, $s(m) = 2^{m^{f(m)}}$ where $f(m) \longrightarrow 0$). Formally,

$$F_d(t(m), \mathrm{PHP}_m, (m+1)m) \wedge P_d(s(m), \Pi_m, t(m), \mathrm{PHP}_m, (m+1)m),$$

where $t(m) = |\mathrm{PHP}_m|$ is a polynomial. Saturation implies that for some non-standard n there exists a depth-d AC^0-Frege proof Π_n of PHP_n of subexponential size, that is

$$F_d(t(n), \mathrm{PHP}_n, (n+1)n) \wedge P_d(s(n), \Pi_n, t(n), \mathrm{PHP}_n, (n+1)n).$$

Furthermore, since $t(n)$ is polynomial and $s(n) = 2^{n^{f(n)}}$ where $f(n)$ is infinitesimal, $t(n), s(n) \in \mathcal{M}_n$.

The formula $F_d(t(n), \mathrm{PHP}_n, (n+1)n) \wedge P_d(s(n), \Pi_n, t(n), \mathrm{PHP}_n, (n+1)n)$ is valid (has truth value $1_{\mathcal{B}}$) in any $K(F, G)$ since it holds in $\mathcal{M}$: this preservation result, which holds for any bounded formula, can be proved by structural induction using the definition of truth. Here $n \in F$ is the constant function given by $n(\omega) = n$, and PHP_n, Π_n are constant strings: for all ω, $(\mathrm{PHP}_n)_\omega$ is the string encoding PHP_n.

The reflection principle implies that $\forall Z S_d(t(n), \mathrm{PHP}_n, (n+1)n, Z)$. The idea is to come up with some $\Delta \in G$ for which $\mathrm{PHP}_n(R)$ is *not* valid (has truth value smaller than $1_{\mathcal{B}}$). Since $\mathrm{PHP}_n(R)$ is equivalent to $S_d(t(n), \mathrm{PHP}_n, (n+1)n, \Delta)$ under V^0, if such a Δ exists then we reach a contradiction, and must conclude that the original proofs of length $s(m)$ cannot have existed. Now let $s(m)$ be the length of the shortest proof of PHP_m. This shows that $s(m)$ cannot be of the form $2^{m^{o(1)}}$, and so $s(m) = 2^{m^{\Omega(1)}}$.

5 A plethora of models

5.1 Shallow decision trees

So far we have described one model, $K(F_{\text{rud}}, G_{\text{rud}})$. While this model satisfies comprehension and induction for *open* formulas (formulas without quantifiers), in order to get a full model of V^0 we need to extend it to a more complicated model $K(F_{\text{tree}}, G_{\text{tree}})$, which is defined as follows.

Let $\delta_0, \delta_1, \ldots$ be rational numbers satisfying the following two properties: $0 < \delta_k < 1$ and $\prod_{i=0}^{\infty} \delta_i > 0$. A *leveled decision tree* is a decision tree of height k in which all nodes at level k are leaves and all nodes at level $i < k$ query a subset $S_i \subset [n]$ of size $n^{\delta_0 \cdots \delta_{i-1}} - n^{\delta_0 \cdots \delta_i}$ depending only on the level i (for $i = 0$, $|S_0| = n - n^{\delta_0}$); furthermore all subsets S_i are disjoint.

Pick an arbitrary non-standard h. The sample space Ω_{tree} consists of pairs $\langle T, \omega \rangle$ where T is a leveled decision tree of height h and $\omega \subseteq [n]$. A function $\alpha \in F_{\text{tree}}$ is given by a standard k and, for each leveled decision tree of height k, a shallow decision tree for every leaf. In order to compute $\alpha(T, \omega)$, truncate the tree T to height k, insert the given shallow decision trees at the leaves, and compute the result on ω. The second sort G_{tree} is defined just like G_{rud}.

The structure $K(F_{\text{tree}}, G_{\text{tree}})$ forms a model of V^0, and the proof crucially requires the Håstad switching lemma. The structure furthermore has some *witnessing* properties and some *preservation* properties.

5.1.1 Witnessing

We start by listing the witnessing properties:

- If $\forall X < x \exists Y < x B(x, X, Y)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$ for some bounded formula $B(x, X, Y)$ then for all standard $\epsilon > 0$ there is $\Gamma \in G_{\text{tree}}$ such that
$$\Pr_{(T,\omega) \in \Omega_{\text{tree}}} [\Gamma_{(T,\omega)} < n \wedge B(n, \omega, \Gamma_{(T,\omega)})] > 1 - \epsilon.$$
- If $\forall X < x \exists Y < x \forall Z < x B(x, X, Y, Z)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$ for some bounded formula $B(x, X, Y, Z)$ then for all standard $\epsilon > 0$ there is $\Gamma \in G_{\text{tree}}$ such that for all $\Theta \in G_{\text{tree}}$,
$$\Pr_{(T,\omega) \in \Omega_{\text{tree}}} [\Gamma_{(T,\omega)} < n \wedge B(n, \omega, \Gamma_{(T,\omega)}, \Theta_{(T,\omega)})] > 1 - \epsilon.$$

These can be used to show that some statements cannot be proved in V^0. For example, let $B(x, X, Y)$ state that for all $i < n$, $Y(i) = X(0) \oplus \cdots \oplus X(i)$. This can be expressed as a bounded formula: $Y(0) = X(0)$ and for all $0 < i < n$, $Y(i) = Y(i-1) \oplus X(i)$. If $\forall X < x \exists Y < x B(x, X, Y)$ were provable in V^0 then it would be valid in $K(F_{\text{tree}}, G_{\text{tree}})$, and so for all standard $\epsilon > 0$ there would exist $\Gamma \in G_{\text{tree}}$ such that
$$\Pr_{(T,\omega) \in \Omega_{\text{tree}}} [\Gamma_{T,\omega} < n \wedge B(n, \omega, \Gamma_{T,\omega})] > 1 - \epsilon.$$

Fix T so that
$$\Pr_{\omega \subseteq [n]} [\Gamma_{T,\omega} < n \wedge B(n, \omega, \Gamma_{T,\omega})] > 1 - \epsilon.$$

We think of ω and $\Gamma_{T,\omega}$ as strings of length n. The event considered states that $\Gamma_{T,\omega}(0) = \omega(0)$ and for all $0 < i < n$, $\Gamma_{T,\omega}(i) = \Gamma_{T,\omega}(i-1) \oplus \omega(i)$.

Suppose that Γ is given by functions $\beta_0, \ldots, \beta_{n-1} \in F_{\text{tree}}$. Then with probability $1 - \epsilon$ over the choice of ω, $\beta_0(T, \omega) = \omega(0)$ and for all $0 < i < n$, $\beta_i(T, \omega) = \beta_{i-1}(T, \omega) \oplus \omega(i)$. The functions $\beta_0, \ldots, \beta_{n-1}$ are all computed (given T) by decision trees of height at most $n - n^{\delta}$ for some standard $\delta > 0$. Let $i < n$ be some element not queried by β_{n-1}. We can prove by induction that if the event holds for both ω and $\omega \vartriangle \{i\}$ (an event that happens with probability $1 - 2\epsilon$) then $\beta_{n-1}(T, \omega) = \beta_{n-1}(T, \omega \Delta \{i\}) \oplus 1$, contradicting the fact that β_{n-1} doesn't query i.

5.1.2 Preservation

The model $K(F_{\text{tree}}, G_{\text{tree}})$ preserves formulas of the following forms:

- $\forall x \forall X < x B(x, X)$, where $B(x, X)$ is a bounded formula.
- $\forall x \forall X < x \exists y < x \forall Z < x B(x, X, y, Z)$, where $B(x, X, y, Z)$ is a bounded formula.
- $\forall x \exists Y < x \forall Z < x B(x, Y, Z)$, where $B(x, Y, Z)$ is a bounded formula.

That is, if one of these formulas is true in $\mathcal{M}$ then it is also true in $K(F_{\text{tree}}, G_{\text{tree}})$. This can be used to prove circuit lower bounds. We start by showing that there is no bounded formula $P(x, X)$ calculating the parity of a string X of length x. If there were such a formula then it would be true in $\mathbf{N}$ (and so in $\mathcal{M}$) that

$$\forall x \forall X < x \forall y < x P(x, X) \leftrightarrow \neg P(x, X \vartriangle \{y\}).$$

(The set $X \Delta \{y\}$ can be defined by a bounded formula.) Preservation of formulas of the first kind implies that this is also true in $K(F_{\text{tree}}, G_{\text{tree}})$.

Let $\Delta \in G_{\text{tree}}$ be given by the functions $\beta_i = 1|_{i \in \omega}$ for all $i < n$. Thus $\Delta_{T,\omega}(i) = \beta_i(T, \omega) = 1|_{i \in \omega}$, in other words, $\Delta_{T,\omega}$ codes the string ω. Substituting $x = n$ and $X = \Delta$, we deduce that

$$\forall y < x P(n, \Delta) \leftrightarrow \neg P(n, \Delta \vartriangle \{y\})$$

is valid in $K(F_{\text{tree}}, G_{\text{tree}})$. Quantifier elimination (described below) shows that for some $\gamma \in F_{\text{tree}}$, $d(P(n, \Delta), \gamma) = 0$, that is the event $P(n, \omega) \neq \gamma(T, \omega)$ has infinitesimal probability. Define a new $\alpha \in F_{\text{tree}}$ by replacing the label of each leaf in the trees defined by γ with some element not queried in the path leading to that leaf. We claim that $P(n, \Delta) \leftrightarrow P(n, \Delta \vartriangle \{\alpha\})$ is valid, leading to a contradiction. Indeed, up to infinitesimal probability, at a point $\langle T, \omega \rangle$ we have $P(n, \Delta) \leftrightarrow \gamma(T, \omega)$ and $P(n, \Delta \vartriangle \{\alpha(T, \omega)\}) \leftrightarrow \gamma(T, \omega \oplus \alpha(T, \omega))$. Since the trees given by γ depend only in T, $\gamma(T, \omega) \leftrightarrow \gamma(T, \omega \oplus \alpha(T, \omega))$.

In order to deduce a circuit lower bound, we follow an argument analogous to the way we obtained a lower bound on lengths of proofs. For any $d \in \mathbf{N}$ there is a bounded formula $V_d(x, X, y, Y)$ which computes the value of the depth-d circuit Y of length y on the input X of length x. For any given m, given a circuit Y_m for parity of size $s(m)$ we can define $P(x, X) = V_d(x, X, s(m), Y)$ for all $x \leq m$. If such circuits exist for all $m \in \mathbf{N}$ then we get a circuit Y_n for some non-standard n, and so can define $P(x, X)$ below n, thus reaching a contradiction. (As stated this gives only a quasipolynomial lower bound since $s(m)$ needs to have subexponential growth; but that can be fixed.)

The two-sorted version of the pigeonhole principle $\text{PHP}(x, R)$ is a bounded formula, and so $\forall x \forall R < x\, \text{PHP}(x, R)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$. Therefore this model cannot be used to prove lower bounds on the pigeonhole principle.

5.2 Algebraic decision trees

The structure $K(F_{\text{tree}}, G_{\text{tree}})$ we have just considered is a model of V^0, and so can be used to reason about AC^0 circuits and AC^0-Frege proofs. We proceed to describe a structure $K(F_{\text{alg}}, G_{\text{alg}})$ which can be used to reason about $\mathsf{AC}^0[2]$ circuits and $\mathsf{AC}^0[2]$-Frege proofs. This time the corresponding theory is Q_2V^0, which is V^0 augmented with the quantifier $Q_2 y < t A(y)$ whose meaning is "$A(y)$ holds an odd number of times in the range $y < t$". This quantifier is axiomatized by the following axioms:

- $\neg Q_2 y < 0 A(y)$.
- $\neg A(t) \to (Q_2 y < t A(y) \leftrightarrow Q_2 y < t + 1 A(y))$.
- $A(t) \to (Q_2 y < t A(y) \leftrightarrow \neg Q_2 y < t + 1 A(y))$.

The added quantifier requires defining the truth value of $[\![Q_2 y < tA(y)]\!]$ so that these axioms hold. The definition is somewhat delicate and omitted for now.

The structure $K(F_{\mathrm{alg}}, G_{\mathrm{alg}})$ has the sample space Ω_{alg} consisting of all subsets of $[n]$ (that is, $\Omega_{\mathrm{alg}} = \Omega_{\mathrm{rud}}$). An *algebraic decision tree* is a decision tree in which internal nodes are labeled by polynomials of low degree (degree n^ϵ for some infinitesimal ϵ), and leaves are labeled with subexponentials. The polynomials are computed modulo 2, and so each internal node has two outgoing edges labeled 0 and 1. The set F_{alg} consists of all functions computed by shallow algebraic decision trees (algebraic decision trees of depth n^ϵ for some infinitesimal ϵ). The set G_{alg} is defined as in the previous cases.

The structure $K(F_{\mathrm{alg}}, G_{\mathrm{alg}})$ is a model of Q_2V^0; in contrast to the previous case, there is no need to extend it to accommodate applications of the switching lemma. The place of the switching lemma is taken by the Razborov-Smolensky method in the following form: a disjunction of an arbitrary number of low degree polynomials can be approximated by a low degree polynomial.

The model $K(F_{\mathrm{alg}}, G_{\mathrm{alg}})$ satisfies those properties of witnessing and preservation enjoyed by $K(F_{\mathrm{tree}}, G_{\mathrm{tree}})$, and as a consequence cannot be used to prove lower bound on the pigeonhole principle. However, witnessing implies that Q_2V^0 cannot prove the existence of the transitive closure of a relation, and preservation can be used to prove lower bounds on circuits computing the mod-p function for odd p.

5.3 PHP decision trees

The structure $K(F_{\mathrm{tree}}, G_{\mathrm{tree}})$, while constituting a model of V^0, satisfies the pigeonhole principle. We now consider a different structure $K(F_{\mathrm{PHP}}, G_{\mathrm{PHP}})$ which is also a model of V^0 but for which the pigeonhole principle fails. As in the case of $K(F_{\mathrm{tree}}, G_{\mathrm{tree}})$, we start by describing a rudimentary model $K(F^0_{\mathrm{PHP}}, G^0_{\mathrm{PHP}})$, and then extend it in a similar fashion to the full-blown model.

The sample space Ω^0_{PHP} of $K(F^0_{\mathrm{PHP}}, G^0_{\mathrm{PHP}})$ consists of all bijections mapping a subset of $[n+1]$ onto $[n]$. A *PHP decision tree* is a decision tree whose internal nodes are labeled with queries of the form $i \mapsto ?$ for $i \in [n+1]$ (with n outgoing edges) and $? \mapsto j$ for $j \in [n]$ (with $n+1$ outgoing edges). As before, leaves are labeled by subexponentials. The set F^0_{PHP} consists of all partial functions computed by shallow PHP decision trees. Members of F^0_{PHP} are only partial functions since $\omega(i)$ could be undefined. However, this happens with infinitesimal probability (as a simple inductive argument shows), and so can be ignored when calculating truth values. The family G^0_{PHP} is defined as in the previous cases.

While $K(F^0_{\mathrm{PHP}}, G^0_{\mathrm{PHP}})$ is not a model of V^0, it is useful as an illustration of a model in which the pigeonhole principle fails. Indeed, consider the function $\Delta^0 \in G^0_{\mathrm{PHP}}$ corresponding to $\beta_{i,j} = 1|_{\omega(i)=j}$ for $i < n+1$ and $j < n$ (clearly $\beta_{i,j} \in F^0_{\mathrm{PHP}}$). We have $\Delta^0_\omega(i,j) = \beta_{i,j}(\omega) = 1|_{\omega(i)=j}$, and so Δ^0_ω is the graph of ω (we consider Δ^0_ω as a two argument function using some pairing function). Therefore, for every $\alpha \in F^0_{\mathrm{PHP}}$ we have

$$\Delta^0(\alpha,\beta)(\omega) = \Delta^0_\omega(\alpha(\omega),\beta(\omega)) = 1|_{\omega(\alpha(\omega))=\beta(\omega)}.$$

We claim that the pigeonhole principle fails for Δ^0. The principle states that one of the following cases holds (to simplify notation, we consider $\Delta^0(i,j)$ as a predicate rather than a $\{0,1\}$ value):

1. $\exists i < n+1 \forall j < n \neg\Delta^0(i,j)$.
2. $\exists i,j < n \exists k < n+1 (i \neq j \wedge \Delta^0(k,i) \wedge \Delta^0(k,j))$.
3. $\exists i,j < n+1 \exists k < n (i \neq j \wedge \Delta^0(i,k) \wedge \Delta^0(j,k))$.

(A weaker principle also allows the possibility that $\exists i < n \forall j < n+1 \neg\Delta^0(j,i)$.) We claim that all these possibilities are invalid, and in fact have truth value $0_{\mathcal{B}}$.

We start with the second statement. Let $\alpha,\beta,\gamma \in F^0_{\mathrm{PHP}}$. We have

$$\begin{aligned}&[\![\alpha,\beta < n \wedge \gamma < n+1 \wedge \alpha \neq \beta \wedge \Delta^0(\gamma,\alpha) \wedge \Delta^0(\gamma,\beta)]\!]\\ =&\langle\!\langle \alpha(\omega),\beta(\omega) < n \wedge \gamma(\omega) < n+1 \wedge \alpha(\omega) \neq \beta(\omega) \wedge \omega(\gamma(\omega)) = \alpha(\omega) \wedge \omega(\gamma(\omega)) = \beta(\omega)\rangle\!\rangle/I = 0_{\mathcal{B}}.\end{aligned}$$

Therefore the second statement has the truth value $0_{\mathcal{B}}$. The third statement similarly has the truth value $0_{\mathcal{B}}$.

We now turn to the first statement. Given $\alpha \in F^0_{\mathrm{PHP}}$, consider $\beta \in F^0_{\mathrm{PHP}}$ which replaces each leaf of α labeled $i < n+1$ with a query $i \mapsto ?$, and labels each leaf by the answer to the query. For this β we have

$$\begin{aligned}&[\![\alpha < n+1 \wedge \beta < n \wedge \neg\Delta^0(\alpha,\beta)]\!]\\ =&\langle\!\langle \alpha(\omega) < n+1 \wedge \beta(\omega) < n \wedge \omega(\alpha(\omega)) \neq \beta(\omega)\rangle\!\rangle/I = 0_{\mathcal{B}}.\end{aligned}$$

This shows that the first statement has the truth value $0_{\mathcal{B}}$ as well. In a similar way, we could extend the argument to the weaker onto pigeonhole principle described above.

5.3.1 Full-blown version

We now define a structure $K(F_{\mathrm{PHP}}, G_{\mathrm{PHP}})$ which is a model of V^0 but in which the pigeonhole principle fails, for the same reason it fails for $K(F^0_{\mathrm{PHP}}, G^0_{\mathrm{PHP}})$. The construction resembles that of $K(F_{\mathrm{tree}}, G_{\mathrm{tree}})$, but is more involved.

Let h be the minimal element of $\mathcal{M}$ satisfying $n^{1/5^h} < 10$. Each point in the sample space Ω_{PHP} consists of a partial injection ω from $[n+1]$ onto $[n]$ together with a partition $R_0, \ldots, R_h$ of $[n]$ into pieces of size $|R_i| = n^{1/5^i} - n^{1/5^{i+1}}$ (where $n^{1/5^{h+1}} = 0$). A function $\alpha \in F_{\mathrm{PHP}}$ is given by an integer $k \in \mathbf{N}$ and a collection of shallow PHP decision trees whose index set is described below. Given $\langle \omega, R_0, \ldots, R_h \rangle \in \Omega_{\mathrm{PHP}}$, α gets to see ω restricted to the inverse image of $R_0 \cup \cdots \cup R_k$, affixes a shallow PHP decision tree depending only on this data, and follows this tree to compute $\alpha(\omega)$. The set G_{PHP} is defined as in the previous cases.

The role of the Håstad switching lemma is now played by the PHP switching lemma, and that explains the appearance of the quantities $n^{1/5^i}$. The counterexample $\Delta \in G_{\mathrm{PHP}}$ is defined analogously to $\Delta^0 \in G^0_{\mathrm{PHP}}$ to satisfy $\Delta(\alpha,\beta)(\omega, R_0, \ldots, R_h) = 1|_{\omega(\alpha(\omega))=\beta(\omega)}$.

6 A glimpse of the proof

The difficult part in showing that a model like $K(F_{\mathrm{tree}}, G_{\mathrm{tree}})$ satisfies V^0 is proving that the comprehension and induction axiom schemes are valid. The strategy is to prove these in two steps. The first, easy step is to prove comprehension and induction for open formulas (formulas without quantifiers); we call these restricted versions *open comprehension* and *open induction*. Then we prove that a version of *quantifier elimination* holds in the model, using the appropriate switching lemma. Comprehension and induction for general bounded formulas then follows.

6.1 Open comprehension and open induction

We start by illustrating how open comprehension and open induction are proved, using the simple model $K(F_{\mathrm{rud}}, G_{\mathrm{rud}})$. Open comprehension states that for every open formula A,

$$\exists\Theta\forall\beta < \alpha(\Theta(\beta) = 1 \leftrightarrow A(\beta, \gamma_1, \ldots, \gamma_k))$$

is valid, where $\alpha, \gamma_1, \ldots, \gamma_k \in F_{\mathrm{rud}}$. The first step is to notice that the leaves in α have some maximal value $m \in \mathcal{M}_n$. Therefore we need to construct for each $i \leq m$ a function δ_i satisfying $\delta_i(\omega) = 1 \leftrightarrow A(i, \gamma_1(\omega), \ldots, \gamma_k(\omega))$. Such functions can be computed by composing the trees corresponding to $\gamma_1, \ldots, \gamma_k$.

Open induction states that for every open formula B,

$$[B(0, \beta_1, \ldots, \beta_k) \wedge \forall\alpha(B(\alpha, \beta_1, \ldots, \beta_k) \to B(\alpha+1, \beta_1, \ldots, \beta_k))] \to \forall\alpha B(\alpha, \beta_1, \ldots, \beta_k),$$

where $\beta_1, \ldots, \beta_k \in F_{\mathrm{rud}}$. Since every $\alpha \in F_{\mathrm{rud}}$ is bounded by some $m \in \mathcal{M}_n$, it is enough to prove the following bounded variant:

$$[B(0, \beta_1, \ldots, \beta_k) \wedge \forall\alpha < m(B(\alpha, \beta_1, \ldots, \beta_k) \to B(\alpha+1, \beta_1, \ldots, \beta_k))] \to B(m, \beta_1, \ldots, \beta_k).$$

Simple manipulation allows us to reduce to the case in which $B(0,\beta_1,\dots,\beta_k)$ and $\neg B(m,\beta_1,\dots,\beta_k)$ are valid (even true for all $\omega \in \Omega_{\mathrm{rud}}$), and then we have to show that

$$\exists \alpha < m(B(\alpha,\beta_1,\dots,\beta_k) \wedge \neg B(\alpha+1,\beta_1,\dots,\beta_k)).$$

We show that this is valid by coming up with $\alpha \in F_{\mathrm{rud}}$ which satisfies this statement. The idea is to use binary search: since we know that $B(0)$ is true and $B(m)$ is false (ignoring the parameters for brevity), if $B(m/2)$ is true then there must be a counterexample in $[m/2,m)$, and otherwise there must be a counterexample in $[0,m)$ (here a counterexample is a value i such that $B(i) \wedge \neg B(i+1)$). Binary search takes logarithmic time, and each step can be implemented using a shallow decision tree. The resulting decision tree has depth $n^\epsilon \log m$ for some infinitesimal ϵ. Since m is subexponential, $\log m = n^\delta$ for some infinitesimal δ, and so the resulting decision tree is also shallow.

6.2 Quantifier elimination

In order to tackle general comprehension and induction, we prove below the following form of quantifier elimination (for $K(F_{\mathrm{tree}},G_{\mathrm{tree}})$). For every open formula $B(x_1,\dots,x_k,y)$ and for all $m \in \mathcal{M}_n$ there exists a function $\Theta \in G_{\mathrm{tree}}$ such that the following formula is valid for all $\alpha_1,\dots,\alpha_k \in F_{\mathrm{tree}}$:

$$[\alpha_1,\dots,\alpha_k < m \wedge \exists\beta < mB(\alpha_1,\dots,\alpha_k,\beta)] \to [\Theta(\alpha_1,\dots,\alpha_k) < m \wedge B(\alpha_1,\dots,\alpha_k,\Theta(\alpha_1,\dots,\alpha_k))]. \tag{1}$$

It then follows that

$$\forall \alpha_1,\dots,\alpha_k < m[\exists\beta < mB(\alpha_1,\dots,\alpha_k,\beta)] \leftrightarrow B(\alpha_1,\dots,\alpha_k,\Theta(\alpha_1,\dots,\alpha_k))$$

is valid. In other words, the formula $\exists\beta < mB(\alpha_1,\dots,\alpha_k,\beta)$ is equivalent to an open formula. Iterating the construction, we get that every bounded formula is equivalent to an open formula below m. Going over the proofs of open comprehension and open induction, we see that this is enough to imply general comprehension and induction.

6.3 Applying the switching lemma

We now complete the proof that $K(F_{\mathrm{tree}},G_{\mathrm{tree}})$ is a model of V^0 by showing that the model admits quantifier elimination in the sense of (1). For simplicity, assume that $k=1$. We want to find for every open formula $B(x,y)$ and subexponential m a function $\Theta \in G_{\mathrm{tree}}$ such that the following is valid:

$$[\alpha < m \wedge \exists\gamma < mB(\alpha,\gamma)] \to [\Theta(\alpha) < m \wedge B(\alpha,\Theta(\alpha))]. \tag{2}$$

The function Θ will be given by functions $\beta_0,\dots,\beta_{m-1} \in F_{\mathrm{tree}}$. Informally, β_i will attempt to find an element j such that $B(i,j)$ holds, if such an element exists. What should the functions $\beta_0,\dots,\beta_{m-1}$ satisfy for (2) to hold? If (2) fails for some $\alpha \in F_{\mathrm{tree}}$ then

$$[\![\Theta(\alpha) < m \wedge B(\alpha,\Theta(\alpha))]\!] < [\![\exists\gamma < mB(\alpha,\gamma)]\!].$$

(We can assume that $\alpha < m$ always holds.) In particular,

$$\mu(\Theta(\alpha) < m \wedge B(\alpha,\Theta(\alpha))) < \mu(\exists\gamma < mB(\alpha,\gamma)).$$

We showed that $\mathcal{B}$ is a complete Boolean algebra by showing that each supremum is equal to a countable supremum. This implies that

$$[\![\exists\gamma < mB(\alpha,\gamma)]\!] = \bigvee_{i\in\mathbf{N}} [\![\gamma_i < m \wedge B(\alpha,\gamma_i)]\!]$$

for some $\gamma_i \in F_{\mathrm{tree}}$. Using the fact that $B(\alpha,i)$ can be evaluated by a shallow decision tree, we can construct for each $k \in \mathbf{N}$ a function $\delta_k \in F_{\mathrm{tree}}$ satisfying

$$[\![\delta_k < m \wedge B(\alpha,\delta_k)]\!] = \bigvee_{i\le k} [\![\gamma_i < m \wedge B(\alpha,\gamma_i)]\!].$$

Therefore $\mu(\delta_k < m \wedge B(\alpha, \delta_k))$ tends to $\mu(\exists \gamma < mB(\alpha, \gamma))$, and so for large enough k,

$$\mu(\Theta(\alpha) < m \wedge B(\alpha, \Theta(\alpha))) < \mu(\delta_k < m \wedge B(\alpha, \delta_k)).$$

This implies that the following set has non-infinitesimal measure:

$$\begin{aligned} &\langle\!\langle \delta_k(\omega) < m \wedge B(\alpha(\omega), \delta_k(\omega)) \rangle\!\rangle \setminus \langle\!\langle \Theta(\alpha)(\omega) < m \wedge B(\alpha(\omega), \Theta(\alpha)(\omega)) \rangle\!\rangle \\ =&\langle\!\langle \delta_k(\omega) < m \wedge B(\alpha(\omega), \delta_k(\omega)) \rangle\!\rangle \setminus \langle\!\langle \beta_{\alpha(\omega)}(\omega) < m \wedge B(\alpha(\omega), \beta_{\alpha(\omega)}(\omega)) \rangle\!\rangle. \end{aligned}$$

In particular,

$$\langle\!\langle \exists y < mB(\alpha(\omega), y) \rangle\!\rangle \setminus \langle\!\langle \beta_{\alpha(\omega)}(\omega) < m \wedge B(\alpha(\omega), \beta_{\alpha(\omega))}(\omega) \rangle\!\rangle$$

has non-infinitesimal measure. One way to guarantee that this never happens is to ensure that

$$\bigcup_{i<m} E_i \triangleq \bigcup_{i<m} \langle\!\langle \exists y < mB(i, y) \rangle\!\rangle \setminus \langle\!\langle \beta_i(\omega) < m \wedge B(i, \beta_i(\omega)) \rangle\!\rangle$$

has infinitesimal measure. (Note that $\langle\!\langle \exists y < mB(i, y) \rangle\!\rangle$ is a non-trivial set since B may have parameters depending on ω.) Below we construct elements $\beta_0, \ldots, \beta_{m-1}$ for which this condition is satisfied.

We start by recalling that

$$\langle\!\langle \exists y < mB(i, y) \rangle\!\rangle = \bigcup_{j<m} \langle\!\langle B(i, j) \rangle\!\rangle.$$

The idea is that β_i will find a $j < m$ satisfying $B(i, j)$ (if any exists) using binary search. The problem is that we need to answer queries of the form "does there exist $j \in [u, v]$ satisfying $B(i, j)$". The range $[u, v]$ could be very big, and so if we combine the decision trees for $B(i, j)$ together the result would not be shallow. The idea is to use the switching lemma in order to approximate the query.

Recall that each $B(i, j)$ is computed by shallow decision trees which replace the nodes at height k of any leveled decision tree (k doesn't depend on i, j). The function β_i will apply one more restriction by replacing nodes at height $k + 1$. Håstad's switching lemma shows that each query can be approximated by a shallow decision tree with a very small probability of error. Hence we can implement the binary search, with only a small probability of error. The probability of error is so small (indeed, exponential) that a union bound shows that the total measure of all E_i is infinitesimal.

6.3.1 Algebraic decision trees

A very similar argument works for algebraic decision trees. We described functions in F_{alg} as computed by shallow algebraic decision trees in which each node is annotated by a low degree polynomial. Each leaf f in such a tree is reached if a certain low degree polynomial p_f is equal to 1, and these polynomials satisfy the additional properties $p_f p_g = 0$ for $f \neq g$ and $\sum_f p_f = 1$. We can use the polynomials p_f as an alternative description of the tree. This shows that every $\{0, 1\}$-valued function in F_{alg} is given by a low degree polynomial.

Following our steps above, we need to find a shallow algebraic decision tree which approximates the predicate $\exists u \leq j \leq vB(i, j)$, which is a disjunction of low degree polynomials. Razborov's trick allows us to approximate this using a single low degree polynomial with low error. The rest of the construction proceeds in the same way.

Since this argument uses an approximation without applying a restriction, we don't require a leveled sample space as in the other two cases.

6.3.2 PHP trees

A similar argument also works in the case of PHP decision trees. The PHP switching lemma replaces Håstad's switching lemma. An added complication is that the functions we consider are only partially defined, and this requires us to be more subtle: while each β_i is undefined with infinitesimal probability, a priori the union of the regions of undefinability can have non-infinitesimal measure. However, since the resulting function is implemented using a shallow PHP decision tree, we know that it is undefined only with infinitesimal probability. (The actual argument is a bit more delicate.)

6.4 Preservation

The main drawback of the model $K(F_{\text{tree}}, G_{\text{tree}})$ is that it preserves too many sentences. We explain why this is the case, and how the model $K(F_{\text{PHP}}, G_{\text{PHP}})$ avoids this.

Suppose that a sentence of the form $\forall x \forall X < x B(x, X)$ holds in $\mathcal{M}$, where $B(x, X)$ is a bounded formula. Take any $\Gamma \in G_{\text{tree}}$. The argument used to prove quantifier elimination shows that for each $m \in \mathcal{M}_n$ there is an open formula $C(x)$ such that for $x < m$, $C(x)$ is equivalent to $B(x, \Gamma)$ in the sense of $K(F_{\text{tree}}, G_{\text{tree}})$ (i.e. $\forall x < m(C(x) \leftrightarrow B(x, \Gamma))$ is valid), and furthermore for all $\alpha \in F_{\text{tree}}$,

$$\langle\!\langle \alpha < m \to B(\alpha, \Gamma) \rangle\!\rangle \vartriangle \langle\!\langle \alpha < m \to C(\alpha) \rangle\!\rangle \in I. \tag{3}$$

Since $\forall x \forall X < x B(x, X)$, $\langle\!\langle \alpha < m \to B(\alpha, \Gamma) \rangle\!\rangle = \Omega_{\text{tree}}$, and so $\mu(\langle\!\langle \alpha < m \to C(\alpha) \rangle\!\rangle) = 1$. Since C is open, $[\![\alpha < m \to C(\alpha)]\!] = \langle\!\langle \alpha < m \to C(\alpha) \rangle\!\rangle / I = 1_{\mathcal{B}}$, and so $\alpha < m \to C(\alpha)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$. Since $C(x)$ is equivalent to $B(x, \Gamma)$ in $K(F_{\text{tree}}, G_{\text{tree}})$ for $x < m$, we deduce that $\forall x < m B(x, \Gamma)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$. Since m was arbitrary, $\forall x \forall X < x B(x, X)$ is valid in $K(F_{\text{tree}}, G_{\text{tree}})$.

This argument fails in $K(F_{\text{PHP}}, G_{\text{PHP}})$ since (3) doesn't hold.