Non-uniform Self-Moduli

Peter Gerdes

Group in Logic
University of California, Berkeley

2007-08 ASL Winter Meeting

Definitions and Notation

Notation

Say $\sigma \succ \tau$ if $\sigma(n) \geq \tau(n)$ everywhere they are both defined.

- So if $f, g \in \omega^\omega$ then $f \succ g \leftrightarrow (\forall n)[f(n) \geq g(n)]$

Definitions

Let $f \in \omega^\omega$ and $X \subset \omega$.

- f is a *modulus (of computation)* for X if for all $g \in \omega^\omega$ if $g \succ f \implies g \geq_T X$.
- f is a *uniform modulus* for X if there is a recursive functional Φ such that $g \succ f \implies \Phi(g) = X$.
- f is a *self-modulus* if f is a modulus for f

Definitions and Notation

Notation

Say $\sigma \succ \tau$ if $\sigma(n) \geq \tau(n)$ everywhere they are both defined.

- So if $f, g \in \omega^\omega$ then $f \succ g \leftrightarrow (\forall n)[f(n) \geq g(n)]$

Definitions

Let $f \in \omega^\omega$ and $X \subset \omega$.

- f is a *modulus (of computation)* for X if for all $g \in \omega^\omega$ if $g \succ f \implies g \geq_T X$.
- f is a *uniform modulus* for X if there is a recursive functional Φ such that $g \succ f \implies \Phi(g) = X$.
- f is a *self-modulus* if f is a modulus for f

Definitions and Notation

Notation

Say $\sigma \succ \tau$ if $\sigma(n) \geq \tau(n)$ everywhere they are both defined.

- So if $f, g \in \omega^\omega$ then $f \succ g \leftrightarrow (\forall n)[f(n) \geq g(n)]$

Definitions

Let $f \in \omega^\omega$ and $X \subset \omega$.

- f is a *modulus (of computation)* for X if for all $g \in \omega^\omega$ if $g \succ f \implies g \geq_T X$.
- f is a *uniform modulus* for X if there is a recursive functional Φ such that $g \succ f \implies \Phi(g) = X$.
- f is a *self-modulus* if f is a modulus for f

Definitions and Notation

Notation

Say $\sigma \succ \tau$ if $\sigma(n) \geq \tau(n)$ everywhere they are both defined.

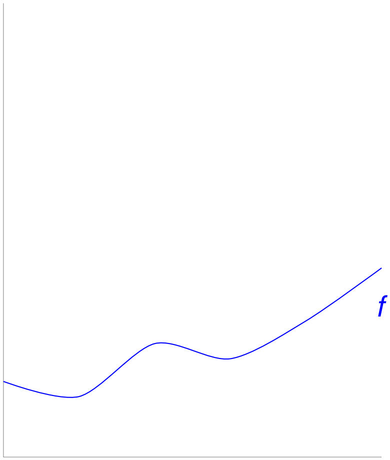
- So if $f, g \in \omega^\omega$ then $f \succ g \leftrightarrow (\forall n)[f(n) \geq g(n)]$

Definitions

Let $f \in \omega^\omega$ and $X \subset \omega$.

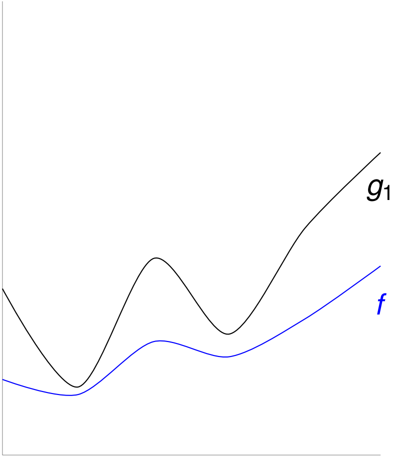
- f is a *modulus (of computation)* for X if for all $g \in \omega^\omega$ if $g \succ f \implies g \geq_T X$.
- f is a *uniform modulus* for X if there is a recursive functional Φ such that $g \succ f \implies \Phi(g) = X$.
- f is a *self-modulus* if f is a modulus for f

Moduli of Computation



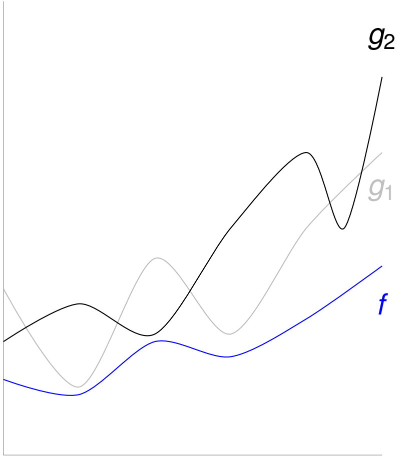
- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Moduli of Computation



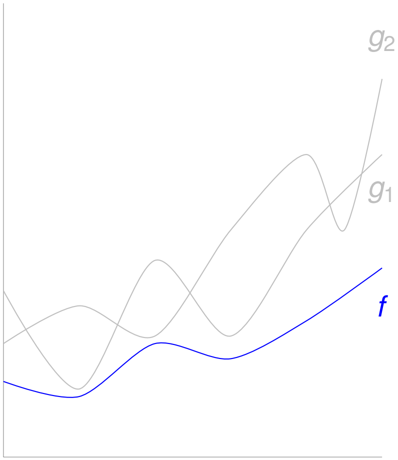
- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Moduli of Computation



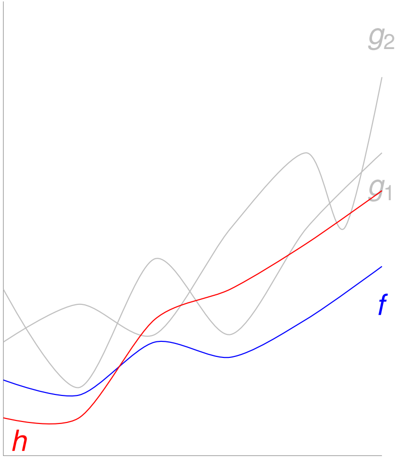
- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Moduli of Computation



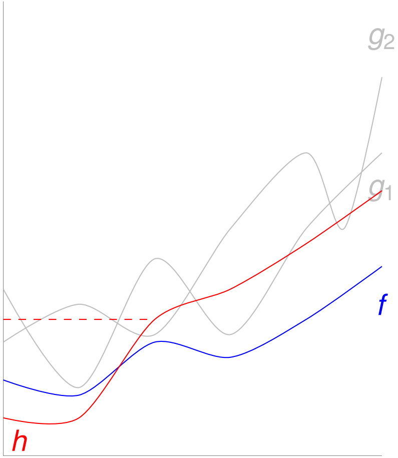
- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Moduli of Computation



- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Moduli of Computation



- Let f be a modulus for X.
- Then $g_1 \succ f \implies g_1 \geq_T X$
- Same with g_2
- f is a uniform modulus if the same reduction works for all $g \succ f$.
- Suppose h is faster growing than f.
- Then h computes X.

Basic Facts

Observation

Every α-REA degree has a uniform self-modulus.

Observation

Every Δ^0_2 degree has a uniform self-modulus.

- Modify proof that Δ^0_2 degrees are hyperimmune.

Theorem (Slaman and Groszek)

There is a uniform self-modulus that computes no non-recursive Δ^0_2-set.

Theorem

For every α there is a uniform self-modulus that computes no non-recursive Δ^0_α-set.

Basic Facts

Observation

Every α-REA degree has a uniform self-modulus.

Observation

Every Δ^0_2 degree has a uniform self-modulus.

- Modify proof that Δ^0_2 degrees are hyperimmune.

Theorem (Slaman and Groszek)

There is a uniform self-modulus that computes no non-recursive Δ^0_2-set.

Theorem

For every α there is a uniform self-modulus that computes no non-recursive Δ^0_α-set.

Basic Facts

Observation

Every α-REA degree has a uniform self-modulus.

Observation

Every Δ^0_2 degree has a uniform self-modulus.

- Modify proof that Δ^0_2 degrees are hyperimmune.

Theorem (Slaman and Groszek)

There is a uniform self-modulus that computes no non-recursive Δ^0_2-set.

Theorem

For every α there is a uniform self-modulus that computes no non-recursive Δ^0_α-set.

Basic Facts

Observation

Every α-REA degree has a uniform self-modulus.

Observation

Every Δ^0_2 degree has a uniform self-modulus.

- Modify proof that Δ^0_2 degrees are hyperimmune.

Theorem (Slaman and Groszek)

There is a uniform self-modulus that computes no non-recursive Δ^0_2-set.

Theorem

For every α there is a uniform self-modulus that computes no non-recursive Δ^0_α-set.

Basic Facts

Observation

Every α-REA degree has a uniform self-modulus.

Observation

Every Δ^0_2 degree has a uniform self-modulus.

- Modify proof that Δ^0_2 degrees are hyperimmune.

Theorem (Slaman and Groszek)

There is a uniform self-modulus that computes no non-recursive Δ^0_2-set.

Theorem

For every α there is a uniform self-modulus that computes no non-recursive Δ^0_α-set.

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^{α}

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

What Degrees Have Moduli?

Theorem (Slaman and Groszek)

X has a modulus if and only if X is Δ^1_1.

Proof.

$\Leftarrow$ $\underset{\sim}{0}^{(\alpha)}$ has a uniform self-modulus. Call it θ^α

$\Rightarrow$ If X has a modulus f then it must also have a uniform modulus $\hat{f}$.

- Try to build $g \succ f$, $g \not\geq_T X$ with Hechler conditions.
- This must fail producing a uniform modulus (and uniform reduction).

A uniform reduction provides a Δ^1_1 definition for X

□

The uniform modulus produced may be very complex

Uniformity

Question

Can we bound the complexity of a uniform modulus for X relative to a modulus for X?

- Sufficent to examine self-moduli.
- Particularly interesting since there is a nice characterization of degrees with uniform self-moduli but not (yet?) for degrees with self-moduli.

Theorem

$\underset{\sim}{d}$ contains a uniform self-modulus iff $\underset{\sim}{d}$ contains a Π^0_2 singleton.

Uniformity

Question

Can we bound the complexity of a uniform modulus for X relative to a modulus for X?

- Sufficent to examine self-moduli.
- Particularly interesting since there is a nice characterization of degrees with uniform self-moduli but not (yet?) for degrees with self-moduli.

Theorem

$\underset{\sim}{d}$ contains a uniform self-modulus iff $\underset{\sim}{d}$ contains a Π^0_2 singleton.

Uniformity

Question

Can we bound the complexity of a uniform modulus for X relative to a modulus for X?

- Sufficent to examine self-moduli.
- Particularly interesting since there is a nice characterization of degrees with uniform self-moduli but not (yet?) for degrees with self-moduli.

Theorem

$\underset{\sim}{d}$ contains a uniform self-modulus iff $\underset{\sim}{d}$ contains a Π^0_2 singleton.

Partial Answer

Theorem

For all $n \in \omega$ there is a self-modulus f so that no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Remark

Going past ω is deceptively hard.

Plan

1. Find a simple property guaranteeing no $h \leq_T f^{(n)}$ is a uniform modulus for f.
2. Build a self-modulus satisfying this property.

Partial Answer

Theorem

For all $n \in \omega$ there is a self-modulus f so that no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Remark

Going past ω is deceptively hard.

Plan

1. Find a simple property guaranteeing no $h \leq_T f^{(n)}$ is a uniform modulus for f.
2. Build a self-modulus satisfying this property.

Partial Answer

Theorem

For all $n \in \omega$ there is a self-modulus f so that no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Remark

Going past ω is deceptively hard.

Plan

1. Find a simple property guaranteeing no $h \leq_T f^{(n)}$ is a uniform modulus for f.
2. Build a self-modulus satisfying this property.

Partial Answer

Theorem

For all $n \in \omega$ there is a self-modulus f so that no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Remark

Going past ω is deceptively hard.

Plan

1. Find a simple property guaranteeing no $h \leq_T f^{(n)}$ is a uniform modulus for f.
2. Build a self-modulus satisfying this property.

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \upharpoonright_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \upharpoonright_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \upharpoonright_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \upharpoonright_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \upharpoonright_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \upharpoonright_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \restriction_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \restriction_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \restriction_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \restriction_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Avoiding Uniformity

Lemma

If f is $n+2$ locally generic on a perfect tree T no $h \leq_T f^{(n)}$ is a uniform modulus for f.

Proof.

- Suppose Φ witnesses $h = \varphi_i(f^{(n)})$ violates the lemma.
- Pick k so $f \restriction_k$ forces both that:
  1. $h = \varphi_i(f^{(n)})$ is total.
  2. If $\sigma \in \omega^{<\omega}$ and $\sigma \succ h$ then $\Phi(\sigma) \subset f$
- Let $\hat{f} \supset f \restriction_k$ be a distinct $n+2$ generic path through T.
- h and $\hat{h}$ must be total so pick $g \succ h, \hat{h}$.
- But $\Phi(g) \subset f$ and $\Phi(g) \subset \hat{f}$ so it can't be total.

□

Guaranteeing Reductions

Uniform Reductions

- Build f computably in $\underset{\sim}{0}^{(n+2)}$
- If $g \succ \theta^{n+2}$ then (uniformly) $g \geq_T f$

How can we guarantee every 'small' $g \succ f$ computes f?

Non-uniformity requires our procedure fails for 'large' g

Idea!

- Use smallness of g to recover f.
- For each $k < n + 2$ encode f into locations f dips below θ^k.
- Since $g \succ f$ we can recover infinitely many of these locations.

Guaranteeing Reductions

Uniform Reductions

- Build f computably in $\underset{\sim}{0}^{(n+2)}$
- If $g \succ \theta^{n+2}$ then (uniformly) $g \geq_T f$

How can we guarantee every 'small' $g \succ f$ computes f?

Non-uniformity requires our procedure fails for 'large' g

Idea!

- Use smallness of g to recover f.
- For each $k < n + 2$ encode f into locations f dips below θ^k.
- Since $g \succ f$ we can recover infinitely many of these locations.

Guaranteeing Reductions

Uniform Reductions

- Build f computably in $\underset{\sim}{0}^{(n+2)}$
- If $g \succ \theta^{n+2}$ then (uniformly) $g \geq_T f$

How can we guarantee every 'small' $g \succ f$ computes f?

Non-uniformity requires our procedure fails for 'large' g

Idea!

- Use smallness of g to recover f.
- For each $k < n + 2$ encode f into locations f dips below θ^k.
- Since $g \succ f$ we can recover infinitely many of these locations.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x-1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \upharpoonright_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x - 1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \upharpoonright_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x-1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \upharpoonright_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x-1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \upharpoonright_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x-1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \upharpoonright_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

First Attempt

Naive Strategy

- Create sequence of trees $T_k \subset T_{k-1}$ for $1 \leq k \leq n+2$ with T_{k+1} representing our attempts to meet Σ^0_{k+1} sets on T_k.
- Prune T_k to ensure at most one $\sigma \in T_{k+1}$ of length $x-1$ satisfies $\sigma(x) < \theta^{k+1}(x)$

- Let k be least such that $g \not\geq_T \underset{\sim}{0}^{(k+1)}$.
- Infinitely often g must dip below θ^{k+1}.
- g can enumerate the set of x with $g(x) < \theta^{k+1}(x)$.
- $f \restriction_x$ is unique $\sigma \in T_{k+1}$ with $\sigma(x) \leq g(x)$.

Problems

1. Need to incorporate multiple strings from T_k in T_{k+1} that aren't above θ^{k+1}

Solution

- $\tau \in T_{k+1}$ must dip below θ^{k+1} for a $\underset{\sim}{0}^{(k)}$-long interval for uniqueness.
- Achieved by 'cancelling' lower priority strings that dip in wrong places.

2. T_{k+1} is a Δ^0_{k+2} set and g only computes $\underset{\sim}{0}^{(k)}$

Solution

$T_{k+1} = \lim_{s\to\infty} T_{k+1}[s]$
Use priority argument to ensure that $g(x)$ is large enough to believe $f\restriction_x \in T_{k+1}$ at true stages.

Problems

1. Need to incorporate multiple strings from T_k in T_{k+1} that aren't above θ^{k+1}

Solution

- $\tau \in T_{k+1}$ must dip below θ^{k+1} for a $\underset{\sim}{0}^{(k)}$-long interval for uniqueness.
- Achieved by 'cancelling' lower priority strings that dip in wrong places.

2. T_{k+1} is a Δ^0_{k+2} set and g only computes $\underset{\sim}{0}^{(k)}$

Solution

$T_{k+1} = \lim_{s \to \infty} T_{k+1}[s]$
Use priority argument to ensure that $g(x)$ is large enough to believe $f \upharpoonright_x \in T_{k+1}$ at true stages.

Problems

1. Need to incorporate multiple strings from T_k in T_{k+1} that aren't above θ^{k+1}

Solution

- $\tau \in T_{k+1}$ must dip below θ^{k+1} for a $\underset{\sim}{0}^{(k)}$-long interval for uniqueness.
- Achieved by 'cancelling' lower priority strings that dip in wrong places.

2. T_{k+1} is a Δ^0_{k+2} set and g only computes $\underset{\sim}{0}^{(k)}$

Solution

$T_{k+1} = \lim_{s\to\infty} T_{k+1}[s]$
Use priority argument to ensure that $g(x)$ is large enough to believe $f \restriction_x \in T_{k+1}$ at true stages.

Problems

1. Need to incorporate multiple strings from T_k in T_{k+1} that aren't above θ^{k+1}

Solution

- $\tau \in T_{k+1}$ must dip below θ^{k+1} for a $\underset{\sim}{0}^{(k)}$-long interval for uniqueness.
- Achieved by 'cancelling' lower priority strings that dip in wrong places.

2. T_{k+1} is a Δ^0_{k+2} set and g only computes $\underset{\sim}{0}^{(k)}$

Solution

$T_{k+1} = \lim_{s \to \infty} T_{k+1}[s]$
Use priority argument to ensure that $g(x)$ is large enough to believe $f \upharpoonright_x \in T_{k+1}$ at true stages.

Problems

1. Need to incorporate multiple strings from T_k in T_{k+1} that aren't above θ^{k+1}

Solution

- $\tau \in T_{k+1}$ must dip below θ^{k+1} for a $\underset{\sim}{0}^{(k)}$-long interval for uniqueness.
- Achieved by 'cancelling' lower priority strings that dip in wrong places.

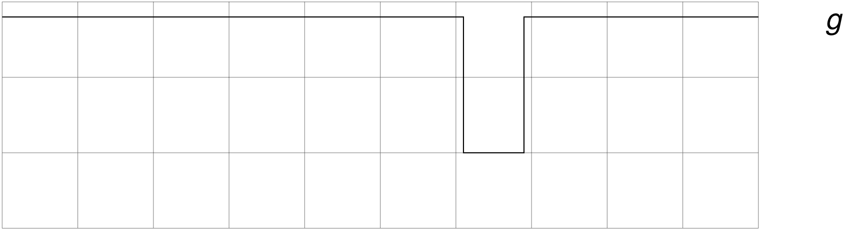
2. T_{k+1} is a Δ^0_{k+2} set and g only computes $\underset{\sim}{0}^{(k)}$

Solution

$T_{k+1} = \lim_{s\to\infty} T_{k+1}[s]$
Use priority argument to ensure that $g(x)$ is large enough to believe $f\restriction_x \in T_{k+1}$ at true stages.

Picturing The Construction

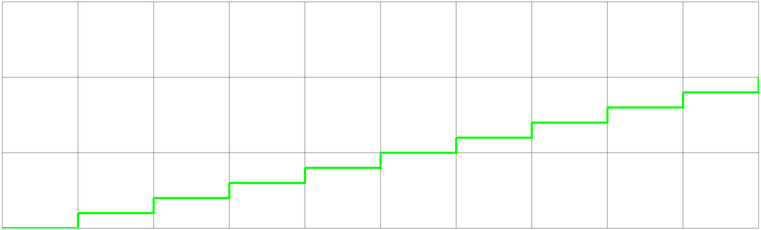
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \upharpoonright_x$.



Function g that wants to compute f

Picturing The Construction

- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \upharpoonright_x$.

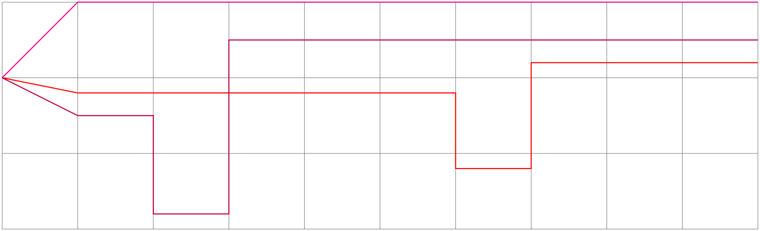


θ^1

Final fast growing function of degree $\underset{\sim}{0}'$.

Picturing The Construction

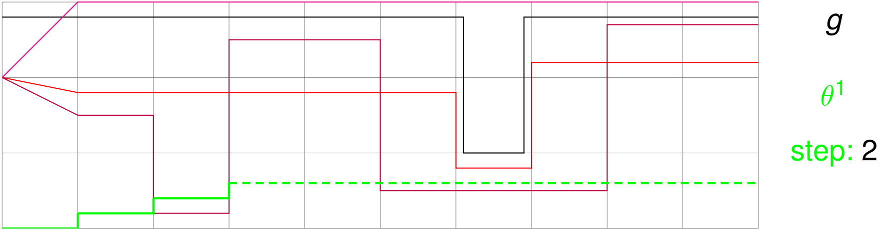
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \restriction_x$.



Final tree T_1 of possible paths for f

Picturing The Construction

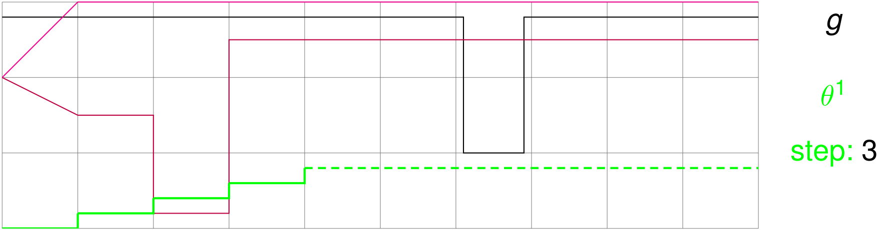
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \upharpoonright_x$.



Membership in T_1 changes during computation steps.

Picturing The Construction

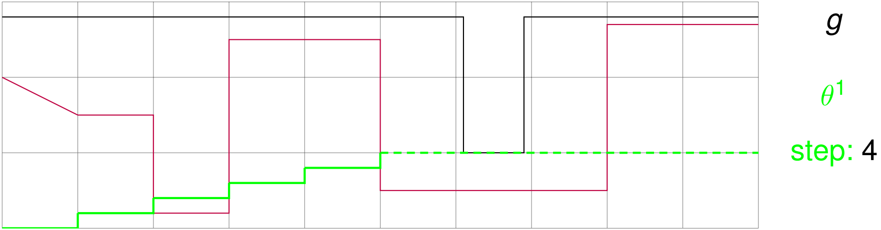
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \upharpoonright_x$.



Membership in T_1 changes during computation steps.

Picturing The Construction

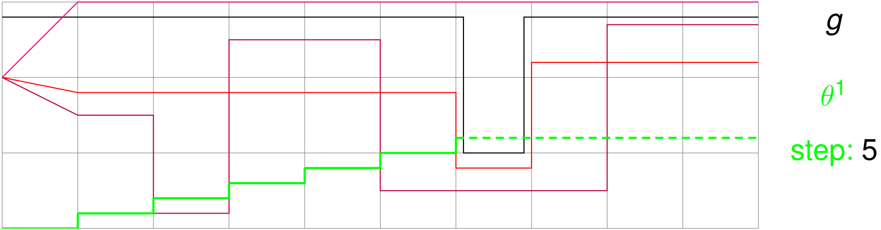
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f\restriction_x$.



Membership in T_1 changes during computation steps.

Picturing The Construction

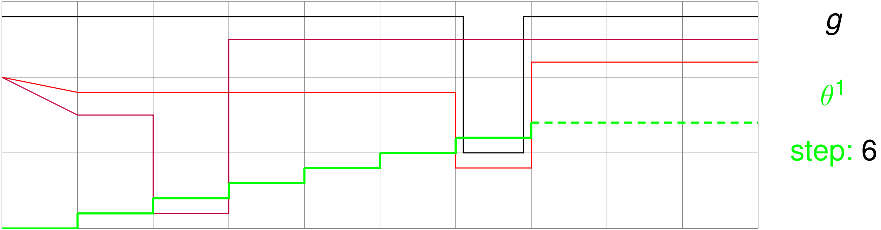
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \restriction_x$.



Membership in T_1 changes during computation steps.

Picturing The Construction

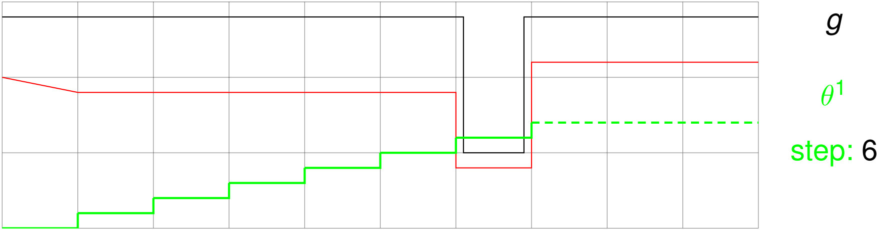
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f\restriction_x$.



At this step g notices a value at which it is small.

Picturing The Construction

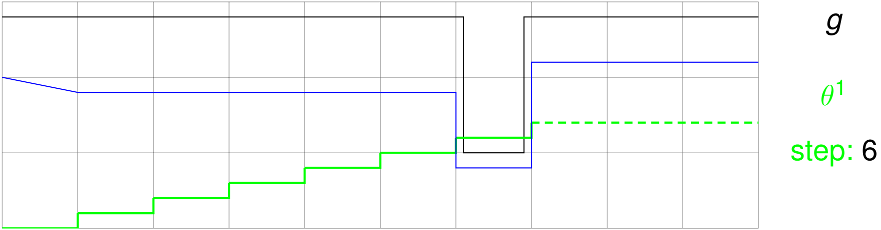
- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \upharpoonright_x$.



Construction guarantees that no false path is below g

Picturing The Construction

- $g \succ f$ searches for stage to commit to f using T_1
- Δ^0_2 guess at T_1 changes with stage.
- Approximation to θ^1 increasing to θ^1
- When $g(x) < \theta^1(x)[s]$ then $T_1[s]$ gives unique value for $f \restriction_x$.



g can commit to an initial segment of f

Thanks

In no particular order:

- My advisor Leo Harrington for taking the time to talk about these issues with me.
- Theodore Slaman for introducing me to moduli of computation.
- The conference organizers for setting this all up.