A bounded jump for the bounded Turing degrees

Bernard Anderson and Barbara Csima

University of Waterloo

October 4, 2010

www.math.uwaterloo.ca/∼b7anders

Computability Theory

Subsets of the natural numbers

In this talk, we will work with subsets of the natural numbers.

Infinite binary strings

We will often identify these sets with infinite binary strings

If $n \in A$ then we say $A(n) = 1$. If $n \notin A$ then we say $A(n) = 0$.

Computability Theory

Subsets of the natural numbers

In this talk, we will work with subsets of the natural numbers.

Infinite binary strings

We will often identify these sets with infinite binary strings

If $n \in A$ then we say $A(n) = 1$. If $n \notin A$ then we say $A(n) = 0$.

We use the notation $A \upharpoonright n$ to denote the elements of A less than or equal to n.

For example, if 3 and 5 are the elements of $A \upharpoonright 5$, then the string for A starts $00101\ldots$

Computability

Computable Sets

We say a set is computable if a sufficiently powerful computer can determine if any number is in the set, given arbitrarily large finite amounts of time and memory space.

Definition of computable

Although the above definition is vague, there are several precise definitions of a set being computable. These definitions have been shown to be equivalent.

Enumerating Turing reductions

Listing programs

We can view programs for our computer as finite strings.

We can use this to find a computable algorithm that lists all possible programs.

Enumerating Turing reductions

Listing programs

We can view programs for our computer as finite strings.

We can use this to find a computable algorithm that lists all possible programs.

Of course, most of these programs will make no sense.

We say a program is total if for every number we input it outputs a number and halts. Otherwise, we say it is partial.

The halting set

Notation

Let φ_n denote the nth program.

If φ_n halts when run with input x we say it converges, denoted $\varphi_n(x)\downarrow$. Otherwise it diverges, denoted $\varphi_n(x)\uparrow$.

The halting set (continued)

The halting set

We define the halting set (also called zero jump) to be the set of numbers n such that the nth program halts when run with input n.

Formally, $\emptyset' = \{x \mid \varphi_x(x)\downarrow\}$.

The halting set (continued)

The halting set

We define the halting set (also called zero jump) to be the set of numbers n such that the nth program halts when run with input n.

Formally, $\emptyset' = \{x \mid \varphi_x(x)\downarrow\}$.

A diagonalization argument can be used to show that $\emptyset'$ is not computable.

Computable Enumerability

Computable Enumerability

We say a set A is computably enumerable (c.e.) if we can computably (effectively) list the elements of the set.

Computable Enumerability

Computable Enumerability

We say a set A is computably enumerable (c.e.) if we can computably (effectively) list the elements of the set.

We can see that $\emptyset'$ is c.e. We enumerate n into $\emptyset'$ when we observe $\varphi_n(n)\downarrow$.

Relative computability

Oracle machines

Let A be a set, and suppose our computer can obtain information about A as part of its computation process.

If this computer can calculate a set B then we say A can compute B.

We denote this $\Phi^A = B$ where Φ is our oracle machine.

Relative computability (continued)

Turing degrees

If A can compute B we also say B is Turing below A and denote it $B \leq_T A$.

Relative computability (continued)

Turing degrees

If A can compute B we also say B is Turing below A and denote it $B \leq_T A$.

We note $\leq_T$ is an equivalence relation, and call the equivalence classes Turing degrees.

The Turing jump

Notation

We let Φ_n^A denote the nth program using A as an oracle.

Turing Jump

We define the Turing jump of A by $A' = \{x \mid \Phi_x^A(x)\downarrow\}$.

The Turing jump

Notation

We let Φ_n^A denote the nth program using A as an oracle.

Turing Jump

We define the Turing jump of A by $A' = \{x \mid \Phi_x^A(x)\downarrow\}$.

As before, a diagonalization argument shows A cannot compute A'. We note A' is c.e. in the oracle A, denoted c.e.(A).

The Turing jump

Notation

We let Φ_n^A denote the nth program using A as an oracle.

Turing Jump

We define the Turing jump of A by $A' = \{x \mid \Phi_x^A(x)\downarrow\}$.

As before, a diagonalization argument shows A cannot compute A'. We note A' is c.e. in the oracle A, denoted c.e.(A).

The Turing jump is one of the most commonly studied items in Computability Theory. We examine its properties.

One to one reductions

1-reductions

We say $A \leq_1 B$ if there is a computable injection $f : \omega \to \omega$ such that $n \in A$ iff $f(n) \in B$.

One to one reductions

1-reductions

We say $A \leq_1 B$ if there is a computable injection $f : \omega \to \omega$ such that $n \in A$ iff $f(n) \in B$.

This is a very strong reduction. $A \leq_1 B$ implies $A \leq_T B$.

We will see it is also stronger than other reducibilities we define later (bT and tt).

Properties of the Turing jump

Basic properties of the Turing jump

Strictly increasing: $A <_T A'$.

Properties of the Turing jump

Basic properties of the Turing jump

Strictly increasing: $A <_T A'$.

Order preserving: $A \leq_T B$ implies $A' \leq_1 B'$.

Properties of the Turing jump

Basic properties of the Turing jump

Strictly increasing: $A <_T A'$.

Order preserving: $A \leq_T B$ implies $A' \leq_1 B'$.

Equivalent to similar forms:
Let $K_0^A = \{\langle x, y \rangle \mid \Phi_x^A(y)\downarrow\}$. Then $A' \equiv_1 K_0^A$.

Properties of the Turing Jump (continued)

Inversion results

- (Friedberg) Let $X \geq_T \emptyset'$. Then there is a Y such that $X \equiv_T Y' \equiv_T Y \oplus \emptyset'$.

Properties of the Turing Jump (continued)

Inversion results

- (Friedberg) Let $X \geq_T \emptyset'$. Then there is a Y such that $X \equiv_T Y' \equiv_T Y \oplus \emptyset'$.

- (Shoenfield) Let $X \geq_T \emptyset'$ be such that X is c.e.($\emptyset'$). Then there is a $Y \leq_T \emptyset'$ such that $X \equiv_T Y'$.

Properties of the Turing Jump (continued)

Inversion results

- (Friedberg) Let $X \geq_T \varnothing'$. Then there is a Y such that $X \equiv_T Y' \equiv_T Y \oplus \varnothing'$.
- (Shoenfield) Let $X \geq_T \varnothing'$ be such that X is c.e.($\varnothing'$). Then there is a $Y \leq_T \varnothing'$ such that $X \equiv_T Y'$.
- (Sacks) The Y in the above result can be made to be c.e.

Arithmetic hierarchy

Σ_1, Π_1, and Δ_1

We say a set A is Σ_1 if there is a computable R such that
$m \in A$ iff $\exists x\ [\langle m, x\rangle \in R]$.

Arithmetic hierarchy

Σ_1, Π_1, and Δ_1

We say a set A is Σ_1 if there is a computable R such that $m \in A$ iff $\exists x\ [\langle m, x\rangle \in R]$.

We say A is Π_1 if $m \in A$ iff $\forall x\ [\langle m, x\rangle \in R]$.

We say a set A is Δ_1 if it is both Σ_1 and Π_1.

Arithmetic hierarchy

Σ_1, Π_1, and Δ_1

We say a set A is Σ_1 if there is a computable R such that $m \in A$ iff $\exists x\ [\langle m, x\rangle \in R]$.

We say A is Π_1 if $m \in A$ iff $\forall x\ [\langle m, x\rangle \in R]$.

We say a set A is Δ_1 if it is both Σ_1 and Π_1.

It can be shown that A is Σ_1 iff A is c.e. and that A is Δ_1 iff A is computable.

Arithmetic hierarchy (continued)

Σ_n, Π_n, and Δ_n

Suppose $m \in A$ iff $Q_1 x_1 \ldots Q_n x_n \; [\langle m, x_1, \ldots, x_n \rangle \in R]$ where $Q_1 \ldots Q_n$ denotes n alternating quantifiers.

If Q_1 is $\exists$ we say A is Σ_n. If Q_n is $\forall$ then A is Π_n.

For example, A is Σ_3 means $m \in A$ iff $\exists x_1 \forall x_2 \exists x_3 \; [\langle m, x_1, x_2, x_3 \rangle \in R]$.

A is Δ_n if it is both Σ_n and Π_n.

Arithmetical hierarchy and the Turing jump

Notation

Let $\overline{A}$ denote the set such that $n \in \overline{A}$ iff $n \notin A$.
Let $\varnothing^{(n)}$ denote the nth Turing jump (i.e. $\varnothing^{(2)}$ is the jump of $\varnothing'$).

Arithmetical hierarchy and the Turing jump

Notation

Let $\overline{A}$ denote the set such that $n \in \overline{A}$ iff $n \notin A$.
Let $\varnothing^{(n)}$ denote the nth Turing jump (i.e. $\varnothing^{(2)}$ is the jump of $\varnothing'$).

Post's Theorem

- A is Δ_n iff $A \leq_T \varnothing^{(n-1)}$.
- A is Σ_n iff A is c.e. $(\varnothing^{(n-1)})$.
- A is Π_n iff $\overline{A}$ is c.e. $(\varnothing^{(n-1)})$.

Other reducibilities

Turing

Recall A is Turing below B if there is an oracle machine which computes A from B.

$A \leq_T B$ if there is a Φ such that $\Phi^A(n) = B(n)$ for all n.

Other reducibilities

Turing

Recall A is Turing below B if there is an oracle machine which computes A from B.

$A \leq_T B$ if there is a Φ such that $\Phi^A(n) = B(n)$ for all n.

Bounded Turing

If we set a computable bound on the amount of the oracle that can be used, we have a bounded Turing reduction.

$A \leq_{bT} B$ if there is a Φ and a computable function f such that $\Phi^{A \upharpoonright f(n)}(n) = B(n)$ for all n.

Other reducibilities (continued)

Truth-table

If we add the requirement that the oracle machine is total, we have truth-table reducibility

$A \leq_{tt} B$ if there is a Φ such that $\Phi^X(n)\downarrow$ for all X and n and $\Phi^A(n) = B$ for all n.

Other reducibilities (continued)

Truth-table

If we add the requirement that the oracle machine is total, we have truth-table reducibility

$A \leq_{tt} B$ if there is a Φ such that $\Phi^X(n)\downarrow$ for all X and n and $\Phi^A(n) = B$ for all n.

Comparing reducibilities

$A \leq_1 B \Rightarrow A \leq_{tt} B \Rightarrow A \leq_{bT} B \Rightarrow A \leq_T B.$

Bounded Turing reducibility is sometimes called weak truth-table reducibility.

The Turing jump on the bounded Turing degrees

Similarities of the Turing jump on the T and bT degrees

We consider the behavior of the Turing jump on the bounded Turing degrees.

Sometimes it acts like the Turing jump on the Turing degrees.

One example where this was discovered to be the case was strong jump inversion.

The Turing jump on the bounded Turing degrees (continued)

Similarities of the T and bT degrees (continued)

Generic reals were used to prove strong jump inversion for the Turing degrees.

(Friedberg) Let $X \geq_T \emptyset'$. Then there is a Y such that $X \equiv_T Y' \equiv_T Y \oplus \emptyset'$.

The Turing jump on the bounded Turing degrees (continued)

Similarities of the T and bT degrees (continued)

Generic reals were used to prove strong jump inversion for the Turing degrees.

(Friedberg) Let $X \geq_T \emptyset'$. Then there is a Y such that $X \equiv_T Y' \equiv_T Y \oplus \emptyset'$.

In 1984 they were used to prove ordinary jump inversion for the truth-table degrees:

(Mohrherr) Let $X \geq_{tt} \emptyset'$. Then there is a Y such that $X \equiv_{tt} Y'$.

The Turing jump on the bounded Turing degrees (continued)

Similarities of the T and bT degrees (continued)

Generic reals cannot be used for strong jump inversion in the tt or bT case.

The Turing jump on the bounded Turing degrees (continued)

Similarities of the T and bT degrees (continued)

Generic reals cannot be used for strong jump inversion in the tt or bT case.

However, newer methods can be used to show strong jump inversion does also hold for the truth-table and bounded Turing degrees.

(Anderson) Let $X \geq_{bT} \emptyset'$. Then there is a Y such that $X \equiv_{bT} Y' \equiv_{bT} Y \oplus \emptyset'$.

The Turing jump on the bounded Turing degrees (continued)

Differences between the T and bT degrees

In other cases the Turing jump acts differently of the bounded Turing degrees than it does on the Turing degrees.

The Turing jump on the bounded Turing degrees (continued)

Differences between the T and bT degrees

In other cases the Turing jump acts differently of the bounded Turing degrees than it does on the Turing degrees.

For example, recall:

(Shoenfield) Let $X \geq_T \emptyset'$ be such that X is c.e.($\emptyset'$). Then there is a $Y \leq_T \emptyset'$ such that $X \equiv_T Y'$.

The Turing jump on the bounded Turing degrees (continued)

Differences between the T and bT degrees

In other cases the Turing jump acts differently of the bounded Turing degrees than it does on the Turing degrees.

For example, recall:

(Shoenfield) Let $X \geq_T \emptyset'$ be such that X is c.e.($\emptyset'$). Then there is a $Y \leq_T \emptyset'$ such that $X \equiv_T Y'$.

The analogue does not hold:

(Csima, Downey, and Ng) There is a $C >_{tt} \emptyset'$ such that C is c.e.($\emptyset'$) but for all $D \leq_T \emptyset'$ we have $D' \not\equiv_{bT} C$.

Motivation

Finding a bounded jump

Can we find a “bounded” jump operator which corresponds to the definition of the bounded Turing degrees?

Motivation

Finding a bounded jump

Can we find a "bounded" jump operator which corresponds to the definition of the bounded Turing degrees?

We would want such an operator to interact with the bounded Turing degrees in a manner analogous to the interaction of the Turing jump with the Turing degrees.

Motivation (continued)

Desired properties

- Limited use of oracle
- Equivalent to similar operators
- Strictly increasing
- Order preserving
- Distinct from known operators

Bounded jump

Defining the bounded jump

We will define the bounded jump to be similar to the Turing jump.

However, we will restrict the use of the oracle for n to the highest possible value of $\varphi_i(n)$ for some $i \leq n$.

Bounded jump (continued)

Definition

$A^b = \{x \mid \exists i < x[\varphi_i(x)\downarrow \ \wedge \Phi_x^{A \upharpoonright \varphi_i(x)}(x)\downarrow]\}.$

We let A^{nb} denote the n-th bounded jump.

Similar operators

A more general form

The bounded jump is equivalent to a more general form.

Definition

$A^{b_0} = \{\langle e, i, j \rangle \mid \varphi_i(j)\downarrow \ \wedge \Phi_e^{A \upharpoonright \varphi_i(j)}(j)\downarrow\}$.

Similar operators

A more general form

The bounded jump is equivalent to a more general form.

Definition

$A^{b_0} = \{\langle e,i,j\rangle \mid \varphi_i(j)\downarrow \ \wedge \Phi_e^{A\upharpoonright\varphi_i(j)}(j)\downarrow\}.$

Theorem

1. $A^{b_0} \leq_1 A^b$
2. $A^b \leq_{tt} A^{b_0}$
3. *There exists A such that $A^b \not\leq_1 A^{b_0}$*

Similar operators (continued)

A simple form

A simplified form does not work as a jump operator.

Definition

$A^i = \{x \mid \Phi_x^{A \upharpoonright x}(x)\downarrow\}$

Remark

Let $A \geq_{bT} \emptyset'$. *Then* $A^i \leq_{bT} A$.

Properties

Basic properties

1. $\emptyset^b \equiv_1 \emptyset'$
2. $A \leq_1 A^b$
3. $A^b \leq_1 A'$ (since A^b is c.e.(A))

Properties (continued)

Strictly increasing

Theorem

$A^b \not\preceq_{bT} A$

Properties (continued)

Strictly increasing

Theorem
$A^b \not\leq_{bT} A$

Order preserving

Theorem
$A \leq_{bT} B \Rightarrow A^{b_0} \leq_1 B^{b_0}$

Properties (continued)

Strictly increasing

Theorem

$A^b \not\leq_{bT} A$

Order preserving

Theorem

$A \leq_{bT} B \Rightarrow A^{b_0} \leq_1 B^{b_0}$

Corollary

1. $A \leq_{bT} B \Rightarrow A^b \leq_{tt} B^b$
2. $\emptyset' \leq_{tt} A^b$

Properties (continued)

A^b and A'

Proposition

$A^b \equiv_T A \oplus \emptyset'$

Properties (continued)

A^b and A'

Proposition

$A^b \equiv_T A \oplus \emptyset'$

Corollary

1. *If* $A' \not\leq_T A \oplus \emptyset'$ *then* $A' \not\leq_T A^b$

Properties (continued)

A^b and A'

Proposition

$A^b \equiv_T A \oplus \emptyset'$

Corollary

1. *If* $A' \not\leq_T A \oplus \emptyset'$ *then* $A' \not\leq_T A^b$
2. *If* $A \geq_T \emptyset'$ *then* $A^b \equiv_T A$

Properties (continued)

A^b and $A \oplus \emptyset'$

Since the bounded jump is strictly increasing, if $A \geq_T \emptyset'$ then $A^b \not\equiv_{bT} A \oplus \emptyset'$

Properties (continued)

A^b and $A \oplus \emptyset'$

Since the bounded jump is strictly increasing, if $A \geq_T \emptyset'$ then $A^b \not\equiv_{bT} A \oplus \emptyset'$

Theorem

The class of A such that $A^b \equiv_{bT} A \oplus \emptyset'$ has measure zero.

Jump inversions

Strong jump inversion

As with the Turing jump, strong jump inversion holds for the bounded jump on the bT degrees.

For every $A \geq_{bT} \emptyset^b$ there is a B such that $B \oplus \emptyset^b \equiv_{bT} B^b \equiv_{bT} A$

Jump inversions

Strong jump inversion

As with the Turing jump, strong jump inversion holds for the bounded jump on the bT degrees.

For every $A \geq_{bT} \emptyset^b$ there is a B such that $B \oplus \emptyset^b \equiv_{bT} B^b \equiv_{bT} A$

Shoenfield jump inversion

We noted earlier that Shoenfield inversion fails to hold for the bounded Turing degrees with the Turing jump.

First main result

Shoenfield jump inversion (continued)

However, Shoenfield inversion does hold for the bounded Turing degrees with the bounded jump.

Theorem

Given B such that $\emptyset^b \leq_{bT} B \leq_{bT} \emptyset^{2b}$ there is an $A \leq_{bT} \emptyset^b$ such that $A^b \equiv_{bT} B$

Second main result (preview)

Arithmetic and Ershov hierarchies

We noted earlier that the Turing jump is closely related to the arithmetic hierarchy.

Similarly, we will show that the bounded jump is closely related to the Ershov hierarchy.

We begin by reviewing the definition of the Ershov hierarchy.

Ershov hierarchy

Computably enumerable (c.e.)

Recall, a set A is c.e. if we can computably list the elements of A.

We can think of this as starting with the empty set, and adding numbers computably (but not removing any).

Ershov hierarchy

Computably enumerable (c.e.)

Recall, a set A is c.e. if we can computably list the elements of A.

We can think of this as starting with the empty set, and adding numbers computably (but not removing any).

2-c.e.

For a 2-c.e. set, we again start with the empty set and can computably add numbers once.

Now, we can also remove numbers (but then we are done, we can't add them back again).

Ershov hierarchy (continued)

m-c.e.

For 2-c.e. sets we are allowed to make at most two changes to a number (one to add it, one to remove it).

For 3-c.e. sets we are allowed to make at most three changes (add, remove, add again).

Similarly, for m-c.e. sets, we are allowed to make at most m changes to a number being in the set.

Ershov hierarchy (continued)

ω-c.e.

A set is ω-c.e. if there is a computable function f such that in deciding if n is in the set, we are allowed to make at most $f(n)$ many changes.

Ershov hierarchy (continued)

ω-c.e.

A set is ω-c.e. if there is a computable function f such that in deciding if n is in the set, we are allowed to make at most $f(n)$ many changes.

We can also use an equivalent definition:

The computable process assigns to each n a number c_n of the maximum number of remaining changes allowed.

Every time a change is made to n being in the set, a lower value must be assigned to c_n.

Ershov hierarchy (continued)

We assign a lexicographic order to ordered pairs of numbers.

For example: $(1, 5) < (1, 72) < (2, 18) < (3, 1) < (3, 15)$ etc.

Ershov hierarchy (continued)

We assign a lexicographic order to ordered pairs of numbers.

For example: $(1,5) < (1,72) < (2,18) < (3,1) < (3,15)$ etc.

ω^2-c.e.

A set is ω^2-c.e. if there is a computable process that assigns to each n an ordered pair c_n.

Every time a change is made to n being in the set, a lower ordered pair must be assigned to c_n.

Ershov hierarchy (continued)

We assign a lexicographic order to ordered pairs of numbers.

For example: $(1,5) < (1,72) < (2,18) < (3,1) < (3,15)$ etc.

ω^2-c.e.

A set is ω^2-c.e. if there is a computable process that assigns to each n an ordered pair c_n.

Every time a change is made to n being in the set, a lower ordered pair must be assigned to c_n.

ω^m-c.e.

Similarly, a set is ω^m-c.e. if the above holds with ordered m-tuples replacing ordered pairs.

Ershov hierarchy (conclusion)

We formalize the definition for arbitrary ordinals.

Definition

A is α-c.e. for $\alpha \geq \omega$ if there is a partial computable $\psi : \omega \times \alpha \to \{0, 1\}$ such that for all n there is a γ such that $\psi(n, \gamma)\downarrow$ and for the least such γ we have $A(n) = \psi(n, \gamma)$.

Ershov hierarchy (conclusion)

We formalize the definition for arbitrary ordinals.

Definition

A is α-c.e. for $\alpha \geq \omega$ if there is a partial computable $\psi : \omega \times \alpha \to \{0, 1\}$ such that for all n there is a γ such that $\psi(n, \gamma)\downarrow$ and for the least such γ we have $A(n) = \psi(n, \gamma)$.

Ershov hierarchy and $\emptyset'$

We note $A \leq_T \emptyset'$ iff A is α-c.e. for some computable ordinal α.

Ershov hierarchy and the bounded jump

We wish to use the bounded jump to characterize the Ershov hierarchy.

To do this, we generalize a well known result.

Second main result

Theorem (Folklore)

$A \leq_{bT} \emptyset' \Leftrightarrow A$ *is* ω*-c.e.* $\Leftrightarrow A \leq_{tt} \emptyset'$

Second main result

Theorem (Folklore)

$A \leq_{bT} \emptyset' \Leftrightarrow A$ is ω-c.e. $\Leftrightarrow A \leq_{tt} \emptyset'$

Theorem

For $n \geq 2$, we have $A \leq_{bT} \emptyset^{nb} \Leftrightarrow A$ is ω^n-c.e. $\Leftrightarrow A \leq_1 \emptyset^{nb}$

tt-cylinders

Definition

A is a tt-cylinder if for all B we have $B \leq_{tt} A \Rightarrow B \leq_1 A$.

Corollary

For $n \geq 2$, we have that $\emptyset^{nb}$ is a tt-cylinder.

Conclusion

Further progress

We can determine if other theorems about the Turing jump hold for the bounded jump.

Conclusion

Further progress

We can determine if other theorems about the Turing jump hold for the bounded jump.

For example, Sacks showed that if B is c.e.($\emptyset'$) and $B \geq_T \emptyset'$ then there is a c.e. set A such that $A' \equiv_T B$.

Csima, Downey, and Ng proved Sacks jump inversion fails for the bounded Turing degrees with the Turing jump.

Not yet known if Sacks jump inversion holds for the bounded Turing degrees with the bounded jump.

Conclusion (continued)

Other open areas

Definition

A is bounded high if $A^b \geq_{bT} \emptyset^{2b}$. A is bounded low if $A^b \leq_{bT} \emptyset^b$.

We can attempt to characterize which sets are bounded high or bounded low. We can also look at other definitions using the bounded jump.

Conclusion (continued)

Other open areas

Definition

A is bounded high if $A^b \geq_{bT} \emptyset^{2b}$. A is bounded low if $A^b \leq_{bT} \emptyset^b$.

We can attempt to characterize which sets are bounded high or bounded low. We can also look at other definitions using the bounded jump.

Gerla developed jump operators for the truth-table and bounded truth-table degrees. Not much work has been done with these operators yet.