1-genericity and the finite intersection principle

Peter Cholak, Rod Downey, Gregory Igusa*

University of Notre Dame

25 June, 2014

In reverse mathematics, we choose a base theory, typically RCA_0, and we analyze the strength of various theorems in terms of what other theorems can be proved from those theorems.

From this point of view, equivalents of the axiom of choice are somehow intrinsically interesting: choice was one of the first principles to be analyzed thoroughly over a base theory, and we seek to understand to what extent the effective analogues of these equivalences continue to hold.

In reverse mathematics, we choose a base theory, typically RCA_0, and we analyze the strength of various theorems in terms of what other theorems can be proved from those theorems.

From this point of view, equivalents of the axiom of choice are somehow intrinsically interesting: choice was one of the first principles to be analyzed thoroughly over a base theory, and we seek to understand to what extent the effective analogues of these equivalences continue to hold.

Definition

A family of sets has the finite intersection property if any finite subfamily has nonempty intersection.

For example, the family of cofinite subsets of ω has the finite intersection property despite having empty intersection.

Theorem (Finite Intersection Principle)

Given any family X of sets, there is a maximal subfamily Y with the finite intersection property.

We bring this principle into second-order arithmetic as follows.

The Finite Intersection Principle

Definition

A family of sets has the finite intersection property if any finite subfamily has nonempty intersection.

For example, the family of cofinite subsets of ω has the finite intersection property despite having empty intersection.

Theorem (Finite Intersection Principle)

Given any family X of sets, there is a maximal subfamily Y with the finite intersection property.

We bring this principle into second-order arithmetic as follows.

The Finite Intersection Principle

Definition

A family of sets has the finite intersection property if any finite subfamily has nonempty intersection.

For example, the family of cofinite subsets of ω has the finite intersection property despite having empty intersection.

Theorem (Finite Intersection Principle)

Given any family X of sets, there is a maximal subfamily Y with the finite intersection property.

We bring this principle into second-order arithmetic as follows.

The Finite Intersection Principle

Definition

A family of sets has the finite intersection property if any finite subfamily has nonempty intersection.

For example, the family of cofinite subsets of ω has the finite intersection property despite having empty intersection.

Theorem (Finite Intersection Principle)

Given any family X of sets, there is a maximal subfamily Y with the finite intersection property.

We bring this principle into second-order arithmetic as follows.

FIP

We use the columns of a real $\mathbb{X}$ to code a family of sets. If we let $X_i = \{\langle i, j\rangle : j \in \mathbb{X}\}$, then we think of $\mathbb{X}$ as coding the family $\{X_i : i \in \omega\}$.

Using this notation, we define the principle FIP.

Definition (Dzhafarov, Mummert)

FIP is the principle of second order arithmetic that states the following:
Given any $\mathbb{X}$, there exists a $\mathbb{Y}$ such that $\{Y_i : i \in \omega\}$ is a maximal subset of $\{X_i : i \in \omega\}$ with the finite intersection property.

We call such an $\mathbb{X}$ an instance of FIP, and we call such a $\mathbb{Y}$ a solution to that instance.

We use the columns of a real $\mathbb{X}$ to code a family of sets. If we let $X_i = \{\langle i, j\rangle : j \in \mathbb{X}\}$, then we think of $\mathbb{X}$ as coding the family $\{X_i : i \in \omega\}$.

Using this notation, we define the principle FIP.

Definition (Dzhafarov, Mummert)

FIP is the principle of second order arithmetic that states the following:
Given any $\mathbb{X}$, there exists a $\mathbb{Y}$ such that $\{Y_i : i \in \omega\}$ is a maximal subset of $\{X_i : i \in \omega\}$ with the finite intersection property.

We call such an $\mathbb{X}$ an instance of FIP, and we call such a $\mathbb{Y}$ a solution to that instance.

FIP

We use the columns of a real $\mathbb{X}$ to code a family of sets. If we let $X_i = \{\langle i, j\rangle : j \in \mathbb{X}\}$, then we think of $\mathbb{X}$ as coding the family $\{X_i : i \in \omega\}$.

Using this notation, we define the principle FIP.

Definition (Dzhafarov, Mummert)

FIP is the principle of second order arithmetic that states the following:
Given any $\mathbb{X}$, there exists a $\mathbb{Y}$ such that $\{Y_i : i \in \omega\}$ is a maximal subset of $\{X_i : i \in \omega\}$ with the finite intersection property.

We call such an $\mathbb{X}$ an instance of FIP, and we call such a $\mathbb{Y}$ a solution to that instance.

In general, most formalizations of the axiom of choice are equivalent to ACA_0 over RCA_0.

Many of them are maximality principles, and it is usually straightforward to code $0'$ into questions of the form "can I add this element to my set?"

However, FIP is an outlier, because we only require that $\mathbb{Y}$ codes a maximal subfamily of $\mathbb{X}$, not a maximal subsequence. So $\mathbb{Y}$ is allowed to "double back" and take columns of $\mathbb{X}$ that it passed over.

Theorem (Dzhafarov, Mummert)

If FIP required that the columns of $\mathbb{Y}$ were a subsequence (not subset) of the columns of $\mathbb{X}$, then FIP would be equivalent to ACA_0 over RCA_0.

ACA_0

In general, most formalizations of the axiom of choice are equivalent to ACA_0 over RCA_0.

Many of them are maximality principles, and it is usually straightforward to code $0'$ into questions of the form "can I add this element to my set?"

However, FIP is an outlier, because we only require that $\mathbb{Y}$ codes a maximal subfamily of $\mathbb{X}$, not a maximal subsequence. So $\mathbb{Y}$ is allowed to "double back" and take columns of $\mathbb{X}$ that it passed over.

Theorem (Dzhafarov, Mummert)

If FIP required that the columns of $\mathbb{Y}$ were a subsequence (not subset) of the columns of $\mathbb{X}$, then FIP would be equivalent to ACA_0 over RCA_0.

ACA_0

In general, most formalizations of the axiom of choice are equivalent to ACA_0 over RCA_0.

Many of them are maximality principles, and it is usually straightforward to code $0'$ into questions of the form "can I add this element to my set?"

However, FIP is an outlier, because we only require that $\mathbb{Y}$ codes a maximal subfamily of $\mathbb{X}$, not a maximal subsequence. So $\mathbb{Y}$ is allowed to "double back" and take columns of $\mathbb{X}$ that it passed over.

Theorem (Dzhafarov, Mummert)

If FIP required that the columns of $\mathbb{Y}$ were a subsequence (not subset) of the columns of $\mathbb{X}$, then FIP would be equivalent to ACA_0 over RCA_0.

ACA_0

In general, most formalizations of the axiom of choice are equivalent to ACA_0 over RCA_0.

Many of them are maximality principles, and it is usually straightforward to code $0'$ into questions of the form "can I add this element to my set?"

However, FIP is an outlier, because we only require that $\mathbb{Y}$ codes a maximal subfamily of $\mathbb{X}$, not a maximal subsequence. So $\mathbb{Y}$ is allowed to "double back" and take columns of $\mathbb{X}$ that it passed over.

Theorem (Dzhafarov, Mummert)

If FIP required that the columns of $\mathbb{Y}$ were a subsequence (not subset) of the columns of $\mathbb{X}$, then FIP would be equivalent to ACA_0 over RCA_0.

1-Generics

In recent work, Diamondstone, Downey, Greenberg, Turetsky find the degree-theoretic principle that we prove is the correct one: computing a Cohen 1-generic.

Observation (DDGT)

Let $\mathbb{X}$ be an instance of FIP.
Let T be the tree of finite subfamilies of $\mathbb{X}$ with nonempty intersection.
Then a 1-generic path through T is a solution to $\mathbb{X}$.

Proof.

Any path through this tree has the finite intersection property.
Genericity ensures that any set that can be added does get added at some point.

1-Generics

In recent work, Diamondstone, Downey, Greenberg, Turetsky find the degree-theoretic principle that we prove is the correct one: computing a Cohen 1-generic.

Observation (DDGT)

Let $\mathbb{X}$ be an instance of FIP.
Let T be the tree of finite subfamilies of $\mathbb{X}$ with nonempty intersection.
Then a 1-generic path through T is a solution to $\mathbb{X}$.

Proof.

Any path through this tree has the finite intersection property. Genericity ensures that any set that can be added does get added at some point.

In recent work, Diamondstone, Downey, Greenberg, Turetsky find the degree-theoretic principle that we prove is the correct one: computing a Cohen 1-generic.

Observation (DDGT)

Let $\mathbb{X}$ be an instance of FIP.
Let T be the tree of finite subfamilies of $\mathbb{X}$ with nonempty intersection.
Then a 1-generic path through T is a solution to $\mathbb{X}$.

Proof.

Any path through this tree has the finite intersection property.
Genericity ensures that any set that can be added does get added at some point. □

1-Generics

In recent work, Diamondstone, Downey, Greenberg, Turetsky find the degree-theoretic principle that we prove is the correct one: computing a Cohen 1-generic.

Observation (DDGT)

Let $\mathbb{X}$ be an instance of FIP.
Let T be the tree of finite subfamilies of $\mathbb{X}$ with nonempty intersection.
Then a 1-generic path through T is a solution to $\mathbb{X}$.

Proof.

Any path through this tree has the finite intersection property. Genericity ensures that any set that can be added does get added at some point. □

Corollary (DDGT)

Any Turing degree that computes a Cohen 1-generic can solve all computable instances of FIP.

Diamondstone, Downey, Greenberg, Turetsky also prove a partial converse:

Theorem (DDGT)

Any Δ^0_2 Turing degree that can solve all computable instances of FIP can compute a 1-generic.

We prove the full converse:

Theorem (Cholak, Downey, I.)

Any Turing degree that can solve all computable instances of FIP can compute a 1-generic.

Corollary (DDGT)

Any Turing degree that computes a Cohen 1-generic can solve all computable instances of FIP.

Diamondstone, Downey, Greenberg, Turetsky also prove a partial converse:

Theorem (DDGT)

Any Δ^0_2 Turing degree that can solve all computable instances of FIP can compute a 1-generic.

We prove the full converse:

Theorem (Cholak, Downey, I.)

Any Turing degree that can solve all computable instances of FIP can compute a 1-generic.

Corollary (DDGT)

Any Turing degree that computes a Cohen 1-generic can solve all computable instances of FIP.

Diamondstone, Downey, Greenberg, Turetsky also prove a partial converse:

Theorem (DDGT)

Any Δ^0_2 Turing degree that can solve all computable instances of FIP can compute a 1-generic.

We prove the full converse:

Theorem (Cholak, Downey, I.)

Any Turing degree that can solve all computable instances of FIP can compute a 1-generic.

The Plan

- We build $\mathbb{X} = \langle X_i \rangle$, a uniformly computable family of subsets of ω.
- Our opponent builds $\mathbb{Y} = \langle Y_i \rangle$, such that $\{Y_i\}$ is a maximal subfamily of $\{X_i\}$ with the finite intersection property.

  We give each set an identifying element, so we may deduce the f such that $Y_i = X_{f(i)}$.
- We use $\mathbb{Y}$ to compute a 1-generic G.

The Plan

- We build $\mathbb{X} = \langle X_i \rangle$, a uniformly computable family of subsets of ω.
- Our opponent builds $\mathbb{Y} = \langle Y_i \rangle$, such that $\{Y_i\}$ is a maximal subfamily of $\{X_i\}$ with the finite intersection property.

  We give each set an identifying element, so we may deduce the f such that $Y_i = X_{f(i)}$.
- We use $\mathbb{Y}$ to compute a 1-generic G.

The Plan

- We build $\mathbb{X} = \langle X_i \rangle$, a uniformly computable family of subsets of ω.
- Our opponent builds $\mathbb{Y} = \langle Y_i \rangle$, such that $\{Y_i\}$ is a maximal subfamily of $\{X_i\}$ with the finite intersection property.

  We give each set an identifying element, so we may deduce the f such that $Y_i = X_{f(i)}$.
- We use $\mathbb{Y}$ to compute a 1-generic G.

The Plan

- We build $\mathbb{X} = \langle X_i \rangle$, a uniformly computable family of subsets of ω.
- Our opponent builds $\mathbb{Y} = \langle Y_i \rangle$, such that $\{Y_i\}$ is a maximal subfamily of $\{X_i\}$ with the finite intersection property.

  We give each set an identifying element, so we may deduce the f such that $Y_i = X_{f(i)}$.
- We use $\mathbb{Y}$ to compute a 1-generic G.

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper.)

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_2 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper.)

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper.)

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper)

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper)

The Framework

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper)

Cats!

- Our construction will be guided by the tree $T = 2^{<\omega}$.
- Our sets X_i will be cats, which will be placed on the tree as follows.
- We place a housecat at every $\sigma \in 2^{<\omega}$.
- There will also be stray cats that can each appear in multiple locations.
  (One stray cat will be used for each Σ^0_1 set that G must either meet or avoid.)
- A finite set of cats will have nonempty intersection if and only if they appear together along some branch of T.

(There are no cats in the printed version of the paper)

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S “sees” infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$.

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S "sees" infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$. Because $\mathbb{Y}$ must be maximal, he must adopt that cat or some other cat that he sees infinitely often.
- We have almost no control over which cat he adopts; we only know that he will adopt at least one stray cat in the limit.

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S "sees" infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$. Because $\mathbb{Y}$ must be maximal, he must adopt that cat or some other cat that he sees infinitely often.
- We have almost no control over which cat he adopts; we only know that he will adopt at least one stray cat in the limit.

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S "sees" infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$. Because $\mathbb{Y}$ must be maximal, he must adopt that cat or some other cat that he sees infinitely often.
- We have almost no control over which cat he adopts, we only know that he will adopt at least one stray cat in the limit.

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S "sees" infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$. Because $\mathbb{Y}$ must be maximal, he must adopt that cat or some other cat that he sees infinitely often.
- We have almost no control over which cat he adopts, we only know that he will adopt at least one stray cat in the limit.

- Some combinatorics shows that our opponent must choose some branch $S \in [T]$, take every cat along that branch, and put it into $\mathbb{Y}$.
- The housecats let us recover S from $\mathbb{Y}$. If it happens that S is 1-generic, we win!
- If S is not 1-generic, then there is at least one stray cat that S "sees" infinitely often.
- Our opponent could adopt that cat into $\mathbb{Y}$. Because $\mathbb{Y}$ must be maximal, he must adopt that cat or some other cat that he sees infinitely often.
- We have almost no control over which cat he adopts, we only know that he will adopt at least one stray cat in the limit.

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Smart cats!

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a family in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Smart cats!

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a family in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Smart cats!

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a family in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Smart cats!

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a family in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Smart cats!

We teach recursion theory to the cats!

- We replace each stray cat with a countable family of stray cats.
- Each family approximates the complement of the halting set in a Π^0_1 manner.
- If our opponent adopts a family at a finite step, then he adopts a computable family. However, if he adopts a family in the *limit* then he only adopts those cats that remain in the family forever.
- So he computes $0'$!

Why 5 years?

Originally the sets were constructed to meet requirements, and the tree was defined implicitly from $\mathbb{Y}$.

From that point of view, it didn't make sense to put more than one set at a given node of the tree.

Also, in order to obtain more control and to do the construction uniformly, after $\mathbb{Y}$ took a "stray cat" that cat would get converted to a house cat in order to force $\mathbb{Y}$ down various paths.

This process forces S to be 1-generic rather than using non-genericity of S for an alternative win.

But you run out of cats if you try to work with more than countably many $\mathbb{Y}'s$.

Reverse Mathematics

Our argument is not uniform, but it is also injury-free.

As a result, we are able to obtain a reverse mathematics result as well.

Theorem (CDI)

$RCA_0 \vdash FIP \leftrightarrow$ 1-gen.

(Finite injury arguments are problematic in reverse mathematics, because they generally require $B\Sigma_2$ to ensure that a finite union of finite injuries is finite.)

Reverse Mathematics

Our argument is not uniform, but it is also injury-free.

As a result, we are able to obtain a reverse mathematics result as well.

Theorem (CDI)

$RCA_0 \vdash FIP \leftrightarrow$ 1-gen.

(Finite injury arguments are problematic in reverse mathematics, because they generally require $B\Sigma_2$ to ensure that a finite union of finite injuries is finite.)

Reverse Mathematics

Our argument is not uniform, but it is also injury-free.

As a result, we are able to obtain a reverse mathematics result as well.

Theorem (CDI)

$RCA_0 \vdash FIP \leftrightarrow$ 1-gen.

(Finite injury arguments are problematic in reverse mathematics, because they generally require $B\Sigma_2$ to ensure that a finite union of finite injuries is finite.)

2IP

Dzhafarov and Mummert also introduce the related notion, 2IP, which says that every family has a maximal subfamily with nonempty pairwise intersection.

They prove that FIP $\Rightarrow$ 2IP, both reverse mathematically and degree theoretically.

Diamondstone, Downey, Greenberg, Turetsky show that FIP and 2IP are equivalent degree theoretic properties in the Δ^0_2 case.

2IP

Dzhafarov and Mummert also introduce the related notion, 2IP, which says that every family has a maximal subfamily with nonempty pairwise intersection.

They prove that FIP $\Rightarrow$ 2IP, both reverse mathematically and degree theoretically.

Diamondstone, Downey, Greenberg, Turetsky show that FIP and 2IP are equivalent degree theoretic properties in the Δ^0_2 case.

Dzhafarov and Mummert also introduce the related notion, 2IP, which says that every family has a maximal subfamily with nonempty pairwise intersection.

They prove that FIP $\Rightarrow$ 2IP, both reverse mathematically and degree theoretically.

Diamondstone, Downey, Greenberg, Turetsky show that FIP and 2IP are equivalent degree theoretic properties in the Δ^0_2 case.

2IP cont

We are able to modify our proof to work for the 2IP case as well:

Theorem (Cholak, Downey, I.)

The 2IP Turing degrees are exactly the FIP Turing degrees, which are also the Turing degrees that bound a 1-generic.

This proof, however, appears to require finite injury in a very intrinsic manner, so reverse-mathematically, we get:

Theorem (CDI)

$RCA_0 + B\Sigma_2 \vdash 2IP \leftrightarrow FIP$

2IP cont

We are able to modify our proof to work for the 2IP case as well:

Theorem (Cholak, Downey, I.)

The 2IP Turing degrees are exactly the FIP Turing degrees, which are also the Turing degrees that bound a 1-generic.

This proof, however, appears to require finite injury in a very intrinsic manner, so reverse-mathematically, we get:

Theorem (CDI)

$RCA_0 + B\Sigma_2 \vdash 2IP \leftrightarrow FIP.$

2IP cont

We are able to modify our proof to work for the 2IP case as well:

Theorem (Cholak, Downey, I.)

The 2IP Turing degrees are exactly the FIP Turing degrees, which are also the Turing degrees that bound a 1-generic.

This proof, however, appears to require finite injury in a very intrinsic manner, so reverse-mathematically, we get:

Theorem (CDI)

$RCA_0 + B\Sigma_2 \vdash 2IP \leftrightarrow FIP.$

2IP cont

We are able to modify our proof to work for the 2IP case as well:

Theorem (Cholak, Downey, I.)

The 2IP Turing degrees are exactly the FIP Turing degrees, which are also the Turing degrees that bound a 1-generic.

This proof, however, appears to require finite injury in a very intrinsic manner, so reverse-mathematically, we get:

Theorem (CDI)

$RCA_0 + B\Sigma_2 \vdash 2IP \leftrightarrow FIP.$

Proof Modification for 2IP

- We must avoid a 1-2, 2-3, 1-3 situation.
- Assign priorities to the families of stray cats.
- Locally injure by initializing an injured family, and then breaking the family into new location-based families that never venture outside of their specific zones.

References

P. Cholak, R. Downey, G. Igusa, Any FIP real computes a 1-generic, submitted

D. Diamondstone, R. Downey, N. Greenberg and D. Turetsky, The finite intersection property and genericity, submitted.

D. Dzhafarov and C. Mummert, Reverse mathematics and properties of finite character, *Annals of Pure and Applied Logic*, Vol. 163, (2012), 1243-1251

End

Thank you