

# Games and Abstract Inductive definitions

Philip Welch

University of Bristol

Kolkata, 5.i.2007

[www.maths.bris.ac.uk/~mapdw](http://www.maths.bris.ac.uk/~mapdw)



- 1) Ordinals and operators.
  - (i) Ordinals
  - (ii) Operators, monotone and non-monotone.
- 2) Circular Definitions (Gupta-Belnap).

Example over  $\mathbb{N}$ , quasi-inductive operators.
- 3) Games
- 4) Closure points of operators and connections with strategies  
Solovay, Svenonius, Martin
- 5) Consequences in weak systems of analysis

The extension into the transfinite of the natural number system:

0 1 2 ...,

The extension into the transfinite of the natural number system:

0 1 2 ...,  $\omega$   $\omega + 1, \dots$

The extension into the transfinite of the natural number system:

$0 \quad 1 \quad 2 \quad \dots, \quad \omega \quad \omega + 1, \dots, \omega + \omega, \dots, \omega^2, \dots, \omega^\omega \dots$

The extension into the transfinite of the natural number system:

$0 \quad 1 \quad 2 \quad \dots, \quad \omega \quad \omega + 1, \dots, \omega + \omega, \dots, \omega^2, \dots, \omega^\omega \dots, \omega_1$

The extension into the transfinite of the natural number system:

$0 \quad 1 \quad 2 \quad \dots, \quad \omega \quad \omega + 1, \dots, \omega + \omega, \dots, \omega^2, \dots, \omega^\omega \dots, \omega_1$

- The *Principle of Transfinite Induction*: a non-empty set of ordinals always has a least member.

The extension into the transfinite of the natural number system:

$0 \quad 1 \quad 2 \quad \dots, \quad \omega \quad \omega + 1, \dots, \omega + \omega, \dots, \omega^2, \dots, \omega^\omega \dots, \omega_1$

- The *Principle of Transfinite Induction*: a non-empty set of ordinals always has a least member.
- This enables schemes of definition by *transfinite recursion*.

# Operators

We consider functions as on sets of integers *operators*.

Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be such.

# Operators

We consider functions as on sets of integers *operators*.

Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be such.

Define by transfinite recursion the following iterates:

$$\Gamma_0(X) = X; \quad \Gamma_{\alpha+1}(X) = \Gamma(\Gamma_\alpha(X)) ;$$

# Operators

We consider functions as on sets of integers *operators*.

Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be such.

Define by transfinite recursion the following iterates:

$$\begin{aligned}\Gamma_0(X) &= X; & \Gamma_{\alpha+1}(X) &= \Gamma(\Gamma_\alpha(X)); \\ \Gamma_\lambda(X) &= \bigcup_{\alpha < \lambda} \Gamma_\alpha(X).\end{aligned}$$

# Operators

We consider functions as on sets of integers *operators*.

Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be such.

Define by transfinite recursion the following iterates:

$$\begin{aligned}\Gamma_0(X) &= X; & \Gamma_{\alpha+1}(X) &= \Gamma(\Gamma_\alpha(X)); \\ \Gamma_\lambda(X) &= \bigcup_{\alpha < \lambda} \Gamma_\alpha(X).\end{aligned}$$

## Definition

An operator  $\Gamma$  is *monotone* if  $X \subseteq Y \longrightarrow \Gamma(X) \subseteq \Gamma(Y)$ .

We consider functions as on sets of integers *operators*.

Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be such.

Define by transfinite recursion the following iterates:

$$\begin{aligned}\Gamma_0(X) &= X; & \Gamma_{\alpha+1}(X) &= \Gamma(\Gamma_\alpha(X)); \\ \Gamma_\lambda(X) &= \bigcup_{\alpha < \lambda} \Gamma_\alpha(X).\end{aligned}$$

## Definition

An operator  $\Gamma$  is *monotone* if  $X \subseteq Y \longrightarrow \Gamma(X) \subseteq \Gamma(Y)$ .

## Definition

For any monotone operator there is a least *countable* ordinal  $\mu = \mu(\Gamma)$  with  $\Gamma_\mu(\emptyset) = \Gamma_{\mu+1}(\emptyset)$ , the *least fixed point*. We write  $\Gamma_\infty$  for  $\Gamma_\mu(\emptyset)$ .

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

$\varphi \in \Sigma_{n+1}^0$  if  $\varphi \equiv \exists \vec{x} \psi$  where  $\psi \in \Pi_n^0$

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

$\varphi \in \Sigma_{n+1}^0$  if  $\varphi \equiv \exists \vec{x} \psi$  where  $\psi \in \Pi_n^0$

$\varphi \in \Pi_{n+1}^0$  if  $\varphi \equiv \neg \psi$  where  $\psi \in \Sigma_{n+1}^0$

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

$\varphi \in \Sigma_{n+1}^0$  if  $\varphi \equiv \exists \vec{x} \psi$  where  $\psi \in \Pi_n^0$

$\varphi \in \Pi_{n+1}^0$  if  $\varphi \equiv \neg \psi$  where  $\psi \in \Sigma_{n+1}^0$

A set  $Y \subseteq \mathbb{N}$  is accordingly  $\Pi_n^0/\Sigma_n^0$  if it can be defined over  $\mathbb{N}$  by a  $\Pi_n^0/\Sigma_n^0$  formula:

$$n \in Y \iff \langle \mathbb{N}, +, \times, 0, S \rangle \models \varphi[n]$$

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

$\varphi \in \Sigma_{n+1}^0$  if  $\varphi \equiv \exists \vec{x} \psi$  where  $\psi \in \Pi_n^0$

$\varphi \in \Pi_{n+1}^0$  if  $\varphi \equiv \neg \psi$  where  $\psi \in \Sigma_{n+1}^0$

A set  $Y \subseteq \mathbb{N}$  is accordingly  $\Pi_n^0/\Sigma_n^0$  if it can be defined over  $\mathbb{N}$  by a  $\Pi_n^0/\Sigma_n^0$  formula:

$$n \in Y \iff \langle \mathbb{N}, +, \times, 0, S \rangle \models \varphi[n]$$

$Y$  is  $\Delta_n^0$  if it is *both*  $\Pi_n^0, \Sigma_n^0$ .

# Classifying operators

We consider a language  $\mathcal{L}_X$  containing a set variable symbol  $\dot{X}$ , extending the usual first order language of arithmetic.

## Definition

The (effective) arithmetic hierarchy:

$\varphi \in \Sigma_0^0$  if  $\varphi \in \mathcal{L}_{\dot{X}}$  contains only bounded number quantifiers.

$\varphi \in \Sigma_{n+1}^0$  if  $\varphi \equiv \exists \vec{x} \psi$  where  $\psi \in \Pi_n^0$

$\varphi \in \Pi_{n+1}^0$  if  $\varphi \equiv \neg \psi$  where  $\psi \in \Sigma_{n+1}^0$

A set  $Y \subseteq \mathbb{N}$  is accordingly  $\Pi_n^0/\Sigma_n^0$  if it can be defined over  $\mathbb{N}$  by a  $\Pi_n^0/\Sigma_n^0$  formula:

$$n \in Y \iff \langle \mathbb{N}, +, \times, 0, S \rangle \models \varphi[n]$$

$Y$  is  $\Delta_n^0$  if it is *both*  $\Pi_n^0, \Sigma_n^0$ .

- Thus an operator  $\Gamma$  may be classified as  $\Sigma_n^0$  if “ $n \in \Gamma(X)$ ” can be written out with a  $\Sigma_n^0$  definition.

If in this definition the symbol  $\dot{X}$  only occurs within an *even* number of negation symbols, then we shall call it a *positive*  $\Sigma_n^0$ -operator. Positiveness guarantees monotonicity.

If in this definition the symbol  $\dot{X}$  only occurs within an *even* number of negation symbols, then we shall call it a *positive*  $\Sigma_n^0$ -operator. Positiveness guarantees monotonicity.

- We may wish to consider more complicated operators: if “ $n \in \Gamma(X)$ ” can be written as a *universal second order* formula

$$\forall f \in \mathbb{N}^{\mathbb{N}} R(f, n, X)$$

(with  $R$  arithmetic) we shall call  $\Gamma$  a  $\Pi_1^1$ -operator.

If in this definition the symbol  $\dot{X}$  only occurs within an *even* number of negation symbols, then we shall call it a *positive*  $\Sigma_n^0$ -operator. Positiveness guarantees monotonicity.

- We may wish to consider more complicated operators: if “ $n \in \Gamma(X)$ ” can be written as a *universal second order* formula

$$\forall f \in \mathbb{N}^{\mathbb{N}} R(f, n, X)$$

(with  $R$  arithmetic) we shall call  $\Gamma$  a  $\Pi_1^1$ -operator.

- A  $\Sigma_1^1$ -operator is defined similarly if “ $n \in \Gamma(X)$ ” can be written using a negated universal (so, equivalently an *existential* second order formula).

# Circular definitions (Gupta-Belnap)

Let<sup>1</sup>  $\mathcal{L}$  be a first order language, suitable for a structure  $\mathcal{M}$  and let  $\mathcal{L}^+$  be its extension by a possibly infinite set of new predicate symbols  $\dot{G}_n(x_1, \dots, x_{k(n)})$ . For each  $\dot{G}$  there is a definition from the set of definitions  $\mathcal{L}$  of the form

$$(1)_n \quad \dot{G}_n(x_1, \dots, x_{k(n)}) =_{df} A_{G_n}(x_1, \dots, x_{k(n)}).$$

---

<sup>1</sup>" *The Revision Theory of Truth*", A. Gupta, N. Belnap, MIT Press, 1993

# Circular definitions (Gupta-Belnap)

Let<sup>1</sup>  $\mathcal{L}$  be a first order language, suitable for a structure  $\mathcal{M}$  and let  $\mathcal{L}^+$  be its extension by a possibly infinite set of new predicate symbols  $\dot{G}_n(x_1, \dots, x_{k(n)})$ . For each  $\dot{G}$  there is a definition from the set of definitions  $\mathcal{L}$  of the form

$$(1)_n \quad \dot{G}_n(x_1, \dots, x_{k(n)}) =_{df} A_{G_n}(x_1, \dots, x_{k(n)}).$$

If we specialise  $\mathcal{M}$  to  $\mathbb{N}$ , and have a single definition arising from a fixed first order formula  $\varphi(v_0, \dot{X})$ , we set:

$$X_{\alpha+1} = \Gamma_{\varphi}(X_{\alpha}) =_{df} \{n \mid \langle \mathbb{N}, +, \times, 0, ', \dots, X_{\alpha} \rangle \models \varphi(n, \dot{X})\}$$

---

<sup>1</sup>"The Revision Theory of Truth", A. Gupta, N. Belnap, MIT Press, 1993

# Circular definitions (Gupta-Belnap)

Let<sup>1</sup>  $\mathcal{L}$  be a first order language, suitable for a structure  $\mathcal{M}$  and let  $\mathcal{L}^+$  be its extension by a possibly infinite set of new predicate symbols  $\dot{G}_n(x_1, \dots, x_{k(n)})$ . For each  $\dot{G}$  there is a definition from the set of definitions  $\mathcal{L}$  of the form

$$(1)_n \quad \dot{G}_n(x_1, \dots, x_{k(n)}) =_{df} A_{G_n}(x_1, \dots, x_{k(n)}).$$

If we specialise  $\mathcal{M}$  to  $\mathbb{N}$ , and have a single definition arising from a fixed first order formula  $\varphi(v_0, \dot{X})$ , we set:

$$X_{\alpha+1} = \Gamma_{\varphi}(X_{\alpha}) =_{df} \{n \mid \langle \mathbb{N}, +, \times, 0, ', \dots, X_{\alpha} \rangle \models \varphi(n, \dot{X})\}$$

The problem here is to decide what to do at limit stages, as there is no question here that such an operator is monotone or even *progressive* (i.e.,  $X \subseteq \Gamma_{\varphi}(X)$ ).

---

<sup>1</sup>"The Revision Theory of Truth", A. Gupta, N. Belnap, MIT Press, 1993

# A Liminf rule and quasi-inductive operators

One proposal is simply that of liminf:

$$X_\lambda = \liminf \langle X_\alpha \mid \alpha < \lambda \rangle =_{\text{df}} \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} X_\beta.$$

# A Liminf rule and quasi-inductive operators

One proposal is simply that of liminf:

$$X_\lambda = \liminf \langle X_\alpha \mid \alpha < \lambda \rangle =_{\text{df}} \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} X_\beta.$$

This gives rise to

## Definition

(i) Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be an operator. We define iterates as before:

$$\Gamma_0(X) = X; \quad \Gamma_{\alpha+1}(X) = \Gamma(\Gamma_\alpha(X));$$

$$\text{For Lim}(\lambda) : \quad \Gamma_\lambda(X) = \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} \Gamma_\beta(X).$$

# A Liminf rule and quasi-inductive operators

One proposal is simply that of liminf:

$$X_\lambda = \liminf \langle X_\alpha \mid \alpha < \lambda \rangle =_{\text{df}} \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} X_\beta.$$

This gives rise to

## Definition

(i) Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be an operator. We define iterates as before:

$$\begin{aligned} \Gamma_0(X) &= X; & \Gamma_{\alpha+1}(X) &= \Gamma(\Gamma_\alpha(X)); \\ \text{For } \text{Lim}(\lambda) : & \Gamma_\lambda(X) &= \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} \Gamma_\beta(X). \end{aligned}$$

(ii) We call  $\Gamma$  an *arithmetic quasi-inductive operator*, if  $\Gamma$  satisfies (i) and is arithmetic.

# A Liminf rule and quasi-inductive operators

One proposal is simply that of liminf:

$$X_\lambda = \liminf \langle X_\alpha \mid \alpha < \lambda \rangle =_{\text{df}} \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} X_\beta.$$

This gives rise to

## Definition

(i) Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be an operator. We define iterates as before:

$$\Gamma_0(X) = X; \quad \Gamma_{\alpha+1}(X) = \Gamma(\Gamma_\alpha(X));$$

$$\text{For } \text{Lim}(\lambda) : \quad \Gamma_\lambda(X) = \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} \Gamma_\beta(X).$$

(ii) We call  $\Gamma$  an *arithmetic quasi-inductive operator*, if  $\Gamma$  satisfies (i) and is arithmetic.

(iii) Let  $\Gamma_\infty = \bigcup_{\alpha \in \text{On}} \bigcap_{\alpha < \beta \in \text{On}} \Gamma_\beta(\emptyset)$  - the *stability set* of  $\Gamma$ .

# A Liminf rule and quasi-inductive operators

One proposal is simply that of liminf:

$$X_\lambda = \liminf \langle X_\alpha \mid \alpha < \lambda \rangle =_{\text{df}} \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} X_\beta.$$

This gives rise to

## Definition

(i) Let  $\Gamma : \mathcal{P}(\mathbb{N}) \rightarrow \mathcal{P}(\mathbb{N})$  be an operator. We define iterates as before:

$$\begin{aligned} \Gamma_0(X) &= X; & \Gamma_{\alpha+1}(X) &= \Gamma(\Gamma_\alpha(X)); \\ \text{For } \text{Lim}(\lambda) : \Gamma_\lambda(X) &= \bigcup_{\alpha < \lambda} \bigcap_{\alpha < \beta < \lambda} \Gamma_\beta(X). \end{aligned}$$

(ii) We call  $\Gamma$  an *arithmetic quasi-inductive operator*, if  $\Gamma$  satisfies (i) and is arithmetic.

(iii) Let  $\Gamma_\infty = \bigcup_{\alpha \in \text{On}} \bigcap_{\alpha < \beta \in \text{On}} \Gamma_\beta(\emptyset)$  - the *stability set* of  $\Gamma$ .

**Remark** There will be a *countable* ordinal  $\zeta = \zeta(\Gamma)$  so that  $\Gamma_\zeta(\emptyset) = \Gamma_\infty$ .

*Examples from Computer Science, philosophical logic*

## Definition

(i)  $Y$  is *inductive* if it is (1-1) reducible to a minimal fixed point of some positive arithmetic operator.

*Examples from Computer Science, philosophical logic*

## Definition

(i)  $Y$  is *inductive* if it is (1-1) reducible to a minimal fixed point of some positive arithmetic operator.

(ii)(Burgess)  $Y$  is *arithmetically quasi-inductive (a.q.i.)* if it is (1-1) reducible to the stability set of some a.q.i.  $\Gamma$ .

*Examples from Computer Science, philosophical logic*

## Definition

(i)  $Y$  is *inductive* if it is (1-1) reducible to a minimal fixed point of some positive arithmetic operator.

(ii)(Burgess)  $Y$  is *arithmetically quasi-inductive (a.q.i.)* if it is (1-1) reducible to the stability set of some a.q.i.  $\Gamma$ .

## Theorem

(i) The  $Y \subseteq \mathbb{N}$  which are  $\Pi_1^1$ , coincide with the inductive sets.

*Examples from Computer Science, philosophical logic*

## Definition

(i)  $Y$  is *inductive* if it is (1-1) reducible to a minimal fixed point of some positive arithmetic operator.

(ii) (Burgess)  $Y$  is *arithmetically quasi-inductive (a.q.i.)* if it is (1-1) reducible to the stability set of some a.q.i.  $\Gamma$ .

## Theorem

(i) The  $Y \subseteq \mathbb{N}$  which are  $\Pi_1^1$ , coincide with the inductive sets.

(ii) (Burgess) The a.q.i.  $Y$  also can be characterized in terms of the Gödel hierarchy of sets.

*Examples from Computer Science, philosophical logic*

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

/ plays  
// plays

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0$   
 $II$  plays

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0$   
 $II$  plays  $k_1$

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0 \quad k_2$   
 $II$  plays  $\quad k_1$

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

I plays  $k_0 \quad k_2$   
II plays  $k_1 \quad k_3$

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0 \quad k_2 \quad \dots k_{2n} \dots$   
 $II$  plays  $k_1 \quad k_3$

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0 \quad k_2 \quad \dots k_{2n} \dots$   
 $II$  plays  $k_1 \quad k_3 \quad \dots k_{2n+1} \dots$

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$I$  plays  $k_0 \quad k_2 \quad \dots k_{2n} \dots$   
 $II$  plays  $k_1 \quad k_3 \quad \dots k_{2n+1} \dots$

- We say  $I$  wins if  $x = (k_0, k_1, \dots) \in A$ .

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$$\begin{array}{l} I \text{ plays } k_0 \quad k_2 \quad \dots k_{2n} \dots \\ II \text{ plays } \quad k_1 \quad k_3 \quad \dots k_{2n+1} \dots \end{array}$$

- We say  $I$  wins if  $x = (k_0, k_1, \dots) \in A$ .
- We say  $G_A$ , or just  $A$ , is *determined* if either player has a winning strategy in this game.

# Infinite 2 person games

- Typically a game between two players is played on integers, or 0's and 1's. Let  $A$  be a set of infinite sequences of natural numbers  $k \in \mathbb{N}$ .
- The game  $G_A$  is defined as follows:

$$\begin{array}{l} I \text{ plays } k_0 \quad k_2 \quad \dots k_{2n} \dots \\ II \text{ plays } \quad k_1 \quad k_3 \quad \dots k_{2n+1} \dots \end{array}$$

- We say  $I$  wins if  $x = (k_0, k_1, \dots) \in A$ .
- We say  $G_A$ , or just  $A$ , is *determined* if either player has a winning strategy in this game.
  - Thus a strategy for player  $I$  is a rule or function that takes the even length sequence  $k_0, \dots, k_{2n-1}$  played so far, and tells him/her what to play for  $k_{2n}$ . Analogously for  $II$ .

## Definition

(1) A *strategy for I* is a function

$$\sigma_I : \{p \in \mathbb{N}^{<\mathbb{N}} \mid lh(p) \text{ is even}\} \rightarrow \mathbb{N}$$

(2) A *strategy for II* is a function

$$\sigma_{II} : \{p \in \mathbb{N}^{<\mathbb{N}} \mid lh(p) \text{ is odd}\} \rightarrow \mathbb{N}.$$

# Strategies (More formally)

## Definition

(1) A *strategy for I* is a function

$$\sigma_I : \{p \in \mathbb{N}^{<\mathbb{N}} \mid \text{lh}(p) \text{ is even}\} \rightarrow \mathbb{N}$$

(2) A *strategy for II* is a function

$$\sigma_{II} : \{p \in \mathbb{N}^{<\mathbb{N}} \mid \text{lh}(p) \text{ is odd}\} \rightarrow \mathbb{N}.$$

## Definition

- 1 A strategy  $\sigma$  for I is a *winning strategy* for  $G(A)$  if I wins each play by using  $\sigma$ . A winning strategy for II is similarly defined.
- 2  $G(A)$  is *determined* if either I or II has a winning strategy for  $G(A)$
- 3 3. If  $\Sigma$  is a class of sets in  $\mathcal{P}(\mathbb{N})$ , then we say  
$$\text{Det}(\Sigma) \iff \forall A \in \Sigma \quad G(A) \text{ is determined.}$$
 (We say that  $G(A)$  is a “ $\Sigma$ -game.”)

# Classification of sets via games and their strategies

## Theorem

*Any  $\Sigma_1^0$ -game has a monotone  $\Pi_1^1$ -inductive strategy (if I wins); if II wins, then a co- $\Pi_1^1$ -inductive strategy can be found.*

# Classification of sets via games and their strategies

## Theorem

*Any  $\Sigma_1^0$ -game has a monotone  $\Pi_1^1$ -inductive strategy (if I wins); if II wins, then a co- $\Pi_1^1$ -inductive strategy can be found.*

## Theorem

*(Solovay, unpublished) Any  $\Sigma_2^0$ -game has a monotone  $\Sigma_1^1$ -inductive strategy (if I wins).*

# Classification of sets via games and their strategies

## Theorem

Any  $\Sigma_1^0$ -game has a monotone  $\Pi_1^1$ -inductive strategy (if I wins); if II wins, then a co- $\Pi_1^1$ -inductive strategy can be found.

## Theorem

(Solovay, unpublished) Any  $\Sigma_2^0$ -game has a monotone  $\Sigma_1^1$ -inductive strategy (if I wins).

We can view the last two theorems (which are theorems in analysis) as provable in *weak subsystems of analysis* ( $\text{RCA}_0 + \text{Determinacy}(\Sigma_1^0)$  and  $\text{RCA}_0 + \text{Determinacy}(\Sigma_2^0)$  respectively).

## Theorem

(Svenonius, Moschovakis) For any  $\Pi_1^1$  set of integers  $Y$ , there is a  $\Sigma_1^0$  set  $U \subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}}$  so that  $Y$  can be represented as:  
 $Y = \{n \mid I \text{ has a w.s. in } G_{(n)}\}$  where  $(n) = \{x \in \mathbb{N}^{\mathbb{N}} \mid (n, x) \in U\}$ .

## Theorem

(Svenonius, Moschovakis) For any  $\Pi_1^1$  set of integers  $Y$ , there is a  $\Sigma_1^0$  set  $U \subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}}$  so that  $Y$  can be represented as:  
 $Y = \{n \mid I \text{ has a w.s. in } G_{(n)}\}$  where  $(n) = \{x \in \mathbb{N}^{\mathbb{N}} \mid (n, x) \in U\}$ .

## Theorem

(Solovay) For any monotone  $\Sigma_1^1$ -inductive set of integers  $Y$ , there is a  $\Sigma_2^0$  set  $U \subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}}$  so that  $Y$  can be represented as:  
 $Y = \{n \mid I \text{ has a w.s. in } G_{(n)}\}$  where  $(n) = \{x \in \mathbb{N}^{\mathbb{N}} \mid (n, x) \in U\}$ .

## Theorem

*On  $\mathcal{P}(\mathbb{N})$ ,  
mon.  $\Pi_1^1$ -inductive  $\subsetneq$  mon.  $\Sigma_1^1$ -inductive  $\subsetneq$  arith. quasi-inductive.*

with the first two corresponding to  $\text{Determinacy}(\Sigma_1^0)$  and  $\text{Determinacy}(\Sigma_2^0)$ .

## Theorem

On  $\mathcal{P}(\mathbb{N})$ ,  
*mon.  $\Pi_1^1$ -inductive  $\subsetneq$  mon.  $\Sigma_1^1$ -inductive  $\subsetneq$  arith. quasi-inductive.*

with the first two corresponding to  $\text{Determinacy}(\Sigma_1^0)$  and  $\text{Determinacy}(\Sigma_2^0)$ .

**Question:** Do  $\Sigma_3^0$  games have a.q.i. strategies?

## Theorem

On  $\mathcal{P}(\mathbb{N})$ ,  
*mon.  $\Pi_1^1$ -inductive  $\subsetneq$  mon.  $\Sigma_1^1$ -inductive  $\subsetneq$  arith. quasi-inductive.*

with the first two corresponding to Determinacy( $\Sigma_1^0$ ) and Determinacy( $\Sigma_2^0$ ).

**Question:** Do  $\Sigma_3^0$  games have a.q.i. strategies?

**Question:** Do a.q.i. sets  $Y$  have game representations as above?

## Theorem

On  $\mathcal{P}(\mathbb{N})$ ,

*mon.  $\Pi_1^1$ -inductive  $\subsetneq$  mon.  $\Sigma_1^1$ -inductive  $\subsetneq$  arith. quasi-inductive.*

with the first two corresponding to  $\text{Determinacy}(\Sigma_1^0)$  and  $\text{Determinacy}(\Sigma_2^0)$ .

**Question:** Do  $\Sigma_3^0$  games have a.q.i. strategies?

**Question:** Do a.q.i. sets  $Y$  have game representations as above?

<u>Det(<math>\Sigma_1^0</math>)</u>	<u>Det(<math>\Sigma_2^0</math>)</u>	<u>??</u>	<u>Det(<math>\Sigma_3^0</math>)</u>
mon. $\Pi_1^1$ -ind.	mon. $\Sigma_1^1$ -ind.	A.Q.I.	???

## Remark

*By a result of H. Friedman (modified by Martin)  $\text{Det}(\Sigma_4^0)$  can not be proven in analysis: it requires  $\text{ZFC}^-$  : the full axioms of Set Theory + AC - Power Set.*

## Remark

*By a result of H. Friedman (modified by Martin)  $\text{Det}(\Sigma_4^0)$  can not be proven in analysis: it requires  $\text{ZFC}^-$  : the full axioms of Set Theory + AC - Power Set.*

It is thus of interest to see if we can use a.q.i. definitions to find strategies of  $\Sigma_3^0$ -games.

## Main Theorem

- (i) Games with payoff sets which are Boolean combinations of  $\Sigma_2^0$  sets, have a.q.i. winning strategies, but*
- (ii) not so for  $\Sigma_3^0$  games.*

Moreover:

### Theorem

*For any arith. quasi-inductive set  $Y$ , there is a  $\Sigma_3^0$  set  $U \subseteq \mathbb{N} \times \mathbb{N}^{\mathbb{N}}$  so that  $Y$  can be represented as:*

*$Y = \{n \mid I \text{ has a w.s. in } G_{(n)}\}$  where  $(n) = \{x \in \mathbb{N}^{\mathbb{N}} \mid (n, x) \in U\}$ .*

If we consider Comprehension Axiom schemes formulated in second order number theory we may relate these to the above notions.

### Theorem

*(i)  $\Pi_3^1$ -Comprehension Axiom proves  $\text{Det}(\Sigma_3^0)$ .*

*(ii)  $\Delta_3^1$ -Comprehension Axiom does not prove  $\text{Det}(\Sigma_3^0)$*

## Theorem

*(Tanaka-MedSalem)  $\Delta_2^1$ -monotone induction +  $\Pi_3^1$ -Transfinite Induction proves  $\text{Det}(\Delta_3^0)$ .*

## Theorem

*(Tanaka-MedSalem)  $\Delta_2^1$ -monotone induction +  $\Pi_3^1$ -Transfinite Induction proves  $\text{Det}(\Delta_3^0)$ .*

## Theorem

*(Tanaka-MedSalem)  $\Delta_3^1$ -Comprehension Axiom does not prove  $\text{Det}(\Delta_3^0)$ .*

Arithmetic quasi-inductive definitions do not tie up exactly with simple arithmetic levels of the determinacy hierarchy.

Arithmetic quasi-inductive definitions do not tie up exactly with simple arithmetic levels of the determinacy hierarchy.

- However their analysis is not without point: one can show that they are intimately involved with levels of the Gödel hierarchy of constructible sets, which are the first building block of a proof-theoretic ordinal analysis of  $\Pi_3^1$ -CA namely links in the level of the  $L$ -hierarchy involving chains of  $\Sigma_2$ -elementary end extensions, just as analysing  $\Pi_2^1$ -CA involves such chains for  $\Sigma_1$  (work of Rathjen, Arai).

Arithmetic quasi-inductive definitions do not tie up exactly with simple arithmetic levels of the determinacy hierarchy.

- However their analysis is not without point: one can show that they are intimately involved with levels of the Gödel hierarchy of constructible sets, which are the first building block of a proof-theoretic ordinal analysis of  $\Pi_3^1$ -CA namely links in the level of the  $L$ -hierarchy involving chains of  $\Sigma_2$ -elementary end extensions, just as analysing  $\Pi_2^1$ -CA involves such chains for  $\Sigma_1$  (work of Rathjen, Arai).

- Hence: if anyone is ever to perform such a proof theoretic analysis, they will first have to analyse the proof theoretic ordinal of a.q.i. definitions.