Building models by games

Ondřej Ježil

March 10, 2021

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).
- The connection to games will not be apparent at first, but the concept will eventually appear in a natural way.

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).
- The connection to games will not be apparent at first, but the concept will eventually appear in a natural way.

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).
- The connection to games will not be apparent at first, but the concept will eventually appear in a natural way.
- 1. A way to build models

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).
- The connection to games will not be apparent at first, but the concept will eventually appear in a natural way.
- 1. A way to build models
  2. Formal treatment of games.

Organization

- We will be going through chapter 2 of Building models by games, (Hodges 1985).
- The connection to games will not be apparent at first, but the concept will eventually appear in a natural way.
- 1. A way to build models
  2. Formal treatment of games.
  3. Forcing with games.

A way of building models

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

- *For every closed L-term t we have* $t = t \in T$.

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

- *For every closed L-term t we have* $t = t \in T$.
- *For every atomic formula* $\phi(x)$ *and all closed L-terms* s, t:
  $(s = t \in T, \phi(t) \in T) \Rightarrow \phi(s) \in T$

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

- *For every closed L-term t we have* $t = t \in T$.
- *For every atomic formula* $\phi(x)$ *and all closed L-terms* s, t:
  $(s = t \in T, \phi(t) \in T) \Rightarrow \phi(s) \in T$

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an =-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ *such that:*

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

- *For every closed L-term t we have* $t = t \in T$.
- *For every atomic formula* $\phi(x)$ *and all closed L-terms* s, t:
  $(s = t \in T, \phi(t) \in T) \Rightarrow \phi(s) \in T$

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an =-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ *such that:*

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$

The canonical model

Definition (=-closed theory)

A first-order L-theory T is =-closed if the following holds:

- *For every closed L-term t we have* $t = t \in T$.
- *For every atomic formula* $\phi(x)$ *and all closed L-terms* s, t:
  $(s = t \in T, \phi(t) \in T) \Rightarrow \phi(s) \in T$

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an =-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ *such that:*

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch)

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed* L*-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic* L*-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch)

- ***Existence***: *Choose* $\{t; t \text{ a closed } L\text{-term}\}/\sim$ *as the underlying set of* $\mathcal{A}$ *where* $t \sim s : \iff (s = t) \in L$*, define the interpretation of functional and relational symbols as:*

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch)

- ***Existence***: *Choose* $\{t; t$ *a closed L-term*$\}/\sim$ *as the underlying set of* $\mathcal{A}$ *where* $t \sim s :\iff (s = t) \in L$, *define the interpretation of functional and relational symbols as:*
  - $f^{\mathcal{A}}([t_1]_\sim, \ldots, [t_n]_\sim) := [f(t_1, \ldots, t_n)]_\sim$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch)

- ***Existence***: *Choose* $\{t;\, t$ *a closed L-term*$\}/\sim$ *as the underlying set of* $\mathcal{A}$ *where* $t \sim s :\iff (s = t) \in L$, *define the interpretation of functional and relational symbols as:*
  - $f^{\mathcal{A}}([t_1]_\sim, \dots, [t_n]_\sim) := [f(t_1, \dots, t_n)]_\sim$
  - $R^{\mathcal{A}}([t_1]_\sim, \dots, [t_n]_\sim) \overset{def}{\iff} R(t_1, \dots, t_n) \in T$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* $: a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* $: \mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch)

- ***Existence***: *Choose $\{t; t$ a closed L-term$\}/\sim$ as the underlying set of $\mathcal{A}$ where $t \sim s :\iff (s = t) \in L$, define the interpretation of functional and relational symbols as:*
  - $f^{\mathcal{A}}([t_1]_\sim, \ldots, [t_n]_\sim) := [f(t_1, \ldots, t_n)]_\sim$
  - $R^{\mathcal{A}}([t_1]_\sim, \ldots, [t_n]_\sim) \overset{def}{\iff} R(t_1, \ldots, t_n) \in T$
- *This is a sound definition because of the $=$-closedness of T.*

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A \, \exists t$ *a closed L-term* $: a = t^{\mathcal{A}}$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch cont.)

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an $=$-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A\, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch cont.)

- ***Uniqueness***: *Notice, that for every $\mathcal{B} \models T$, there is a unique homomorphism $f : \mathcal{A} \to \mathcal{B}$ sending $t_\sim \mapsto t^{\mathcal{B}}$, this follows just from the two requirements we have one $\mathcal{A}$.*

Canonical model cont.

Theorem (Canonical model theorem)

Let L contain at least one constant and let T be an =-closed L-theory which contains only atomic sentences, then there exists a structure $\mathcal{A}$ such that:

- $\forall a \in A \, \exists t$ *a closed L-term* : $a = t^{\mathcal{A}}$
- $\forall \phi$ *an atomic L-sentence* : $\mathcal{A} \models \phi \iff \phi \in T$

Proof (sketch cont.)

- ***Uniqueness***: *Notice, that for every $\mathcal{B} \models T$, there is a unique homomorphism $f : \mathcal{A} \to \mathcal{B}$ sending $t_{\sim} \mapsto t^{\mathcal{B}}$, this follows just from the two requirements we have one $\mathcal{A}$.*
- *So if $\mathcal{A}'$ were satisfy the statement of the theorem we would have $g : \mathcal{A} \to \mathcal{A}'$ and $h : \mathcal{A}' \to \mathcal{A}$, so $h \circ g : \mathcal{A} \to \mathcal{A}$ but only such homomorphism is the identity.* □

Notions of consistency

- We would like to formalize sufficient condition for a theory to have a model.

Notions of consistency

- We would like to formalize sufficient condition for a theory to have a model.
- We can do so without proof calculus using the following notion.

Notions of consistency

- We would like to formalize sufficient condition for a theory to have a model.
- We can do so without proof calculus using the following notion.
- You can think of it like this: What are the conditions for some theory such that the theory “could be true in some structure”?

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p \text{ and } (t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$*
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$*
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula* $\phi(x)$ *and s, t closed L-terms:* $(\phi(t) \in p \text{ and } (t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every* $p \in N$ *and every L-sentence* ϕ*: both* ϕ *and* $\neg\phi$ *cannot be in* p
4. *for every* $p \in N$ *and every L-sentence* ϕ*: both* ϕ *and* $\neg\phi$ *cannot be in* p

Now for ϕ *and* ψ *L-sentences:*

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p \text{ and } (t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p
4. for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

For all variables x and L-formulas $\phi(x)$:

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ for every closed L-term t

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula* $\phi(x)$ *and* s, t *closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every* $p \in N$ *and every L-sentence* ϕ*: both* ϕ *and* $\neg\phi$ *cannot be in* p
4. *for every* $p \in N$ *and every L-sentence* ϕ*: both* ϕ *and* $\neg\phi$ *cannot be in* p

Now for ϕ *and* ψ *L-sentences:*

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

For all variables x *and L-formulas* $\phi(x)$*:*

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for every closed L-term* t
10. $\neg\forall x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for some closed L-term* t

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$*
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

For all variables x and L-formulas $\phi(x)$:

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for every closed L-term t*
10. $\neg\forall x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for some closed L-term t*
11. $\exists x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for some closed L-term t*

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$*
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

For all variables x and L-formulas $\phi(x)$:

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for every closed L-term t*
10. $\neg\forall x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for some closed L-term t*
11. $\exists x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for some closed L-term t*
12. $\neg\exists x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for every closed L-term t*

Notions of consistency cont.

Definition (Notion of consistency)

Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a ***notion of consistency*** *if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t:* $p \in N \Rightarrow p \cup \{t = t\} \in N$
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms:* $(\phi(t) \in p$ *and* $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

For all variables x and L-formulas $\phi(x)$:

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for every closed L-term t*
10. $\neg\forall x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for some closed L-term t*
11. $\exists x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for some closed L-term t*
12. $\neg\exists x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for every closed L-term t*

Notions of consistency cont.

Definition (Notion of consistency)

*Let L be a first-order language of cardinality λ. We shall call a set of sets of L-sentences N a **notion of consistency** if the following conditions hold for every set $p \in N$:*

1. *for every closed L-term t: $p \in N \Rightarrow p \cup \{t = t\} \in N$*
2. *for every atomic L-formula $\phi(x)$ and s, t closed L-terms: $(\phi(t) \in p$ and $(t \overset{\leftrightarrow}{=} s) \in p) \Rightarrow p \cup \{\phi(s)\} \in N$*
3. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*
4. *for every $p \in N$ and every L-sentence ϕ: both ϕ and $\neg\phi$ cannot be in p*

Now for ϕ and ψ L-sentences:

5. $\phi \wedge \psi \in p \Rightarrow p \cup \{\phi, psi\} \in N$
6. $\neg(\phi \wedge \psi) \in p \Rightarrow \{p \cup \{\neg\phi\}, p \cup \{\neg\psi\}\} \cap N \neq \varnothing$
7. $\phi \vee \psi \in p \Rightarrow \{p \cup \{\phi\}, p \cup \{\psi\}\} \cap N \neq \varnothing$
8. $\neg(\phi \vee \psi) \in p \Rightarrow p \cup \{\neg\psi, \neg\phi\} \in N$

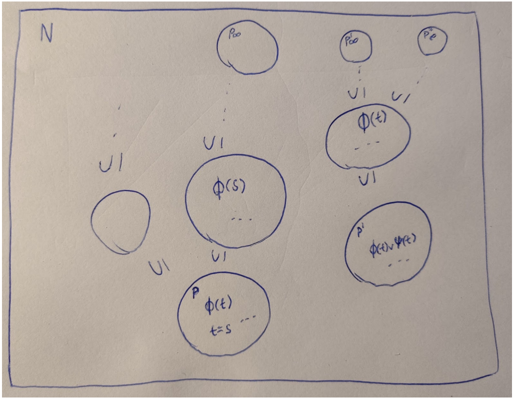
For all variables x and L-formulas $\phi(x)$:

9. $\forall x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for every closed L-term t*
10. $\neg\forall x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for some closed L-term t*
11. $\exists x : \phi(x) \in p \Rightarrow p \cup \{\phi(t)\} \in N$ *for some closed L-term t*
12. $\neg\exists x : \phi(x) \in p \Rightarrow p \cup \{\neg\phi(t)\} \in N$ *for every closed L-term t*

Finally:

13. *if $\alpha < \lambda$, $(p_i)_{i<\lambda}$ is an increasing chain in N, $|\bigcup_{i<\alpha} p_i \setminus p_0| < \lambda$, then $(\bigcup_{i<\alpha} p_i) \in N$.*

A picture!



Remarks

- Intuitively having a notion of consistency mean that we can unpack sentences in a theory into atomic sentences.

Remarks

- Intuitively having a notion of consistency mean that we can unpack sentences in a theory into atomic sentences.
- Notice that for some consistent theories we don't necessarily have a way to construct a notion of consistency, since the language L may not have enough terms, we can however always enrich the language. (As in the proof of the completeness theorem.)

Remarks

- Intuitively having a notion of consistency mean that we can unpack sentences in a theory into atomic sentences.
- Notice that for some consistent theories we don't necessarily have a way to construct a notion of consistency, since the language L may not have enough terms, we can however always enrich the language. (As in the proof of the completeness theorem.)
- Elements $p \in N$ are called **conditions**.

A notion of consistency implies satisfiability

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

A notion of consistency implies satisfiability

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof

Let $p_0 := p$. We will construct a λ-long ascending chain of conditions $(p_i)_{i<\lambda}$ using the properties of N. At each step we will only add finitely many new sentences so by (13) we have that $\bigcup \overline{p} := \bigcup_{i<\lambda} p_i \in N$.

A notion of consistency implies satisfiability

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof

Let $p_0 := p$. We will construct a λ-long ascending chain of conditions $(p_i)_{i<\lambda}$ using the properties of N. At each step we will only add finitely many new sentences so by (13) we have that $\bigcup \overline{p} := \bigcup_{i<\lambda} p_i \in N$.

A notion of consistency implies satisfiability

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof

Let $p_0 := p$. We will construct a λ-long ascending chain of conditions $(p_i)_{i<\lambda}$ using the properties of N. At each step we will only add finitely many new sentences so by (13) we have that $\bigcup \overline{p} := \bigcup_{i<\lambda} p_i \in N$.
How to construct the chain? We will use tasks that correspond to the instances of properties (1),(2),(4)-(12) of N. For example task corresponding to (7) and formulas ϕ, ψ is:

$$\text{If } \phi \vee \psi \in \bigcup \overline{p}\text{, put at least one of } \phi,\ \psi \text{ into } \bigcup \overline{p}.$$

A notion of consistency implies satisfiability

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

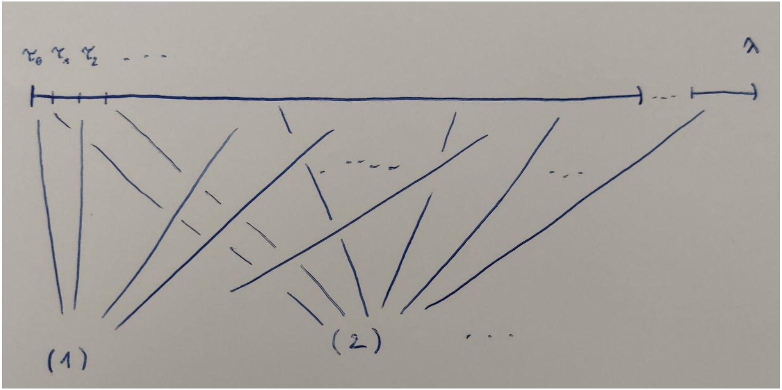
Proof

Let $p_0 := p$. We will construct a λ-long ascending chain of conditions $(p_i)_{i<\lambda}$ using the properties of N. At each step we will only add finitely many new sentences so by (13) we have that $\bigcup \overline{p} := \bigcup_{i<\lambda} p_i \in N$.
How to construct the chain? We will use tasks that correspond to the instances of properties (1),(2),(4)-(12) of N. For example task corresponding to (7) and formulas ϕ, ψ is:

$$\text{If } \phi \vee \psi \in \bigcup \overline{p}, \text{ put at least one of } \phi,\ \psi \text{ into } \bigcup \overline{p}.$$

Organize the tasks into a sequence $(\tau_i)_{i<\lambda}$ such that for each property we have λ many instances and the task as a whole is completed by the time λ.

A picture for the task organization



A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.
Let $U \subseteq \bigcup \overline{p}$ be the subset of all atomic sentences. By the tasks corresponding to (1) *and* (2) *we have that U is $=$-closed and so by the Canonical model theorem we have $\mathcal{A} \models U$.*

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.
Let $U \subseteq \bigcup \overline{p}$ be the subset of all atomic sentences. By the tasks corresponding to (1) *and* (2) *we have that U is $=$-closed and so by the Canonical model theorem we have $\mathcal{A} \models U$.*
Claim: *For every L-sentence ϕ we have:*

$$\phi \in \bigcup \overline{p} \Rightarrow \mathcal{A} \models \phi.$$

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.
Let $U \subseteq \bigcup \overline{p}$ be the subset of all atomic sentences. By the tasks corresponding to (1) *and* (2) *we have that U is $=$-closed and so by the Canonical model theorem we have $\mathcal{A} \models U$.*
Claim: *For every L-sentence ϕ we have:*

$$\phi \in \bigcup \overline{p} \Rightarrow \mathcal{A} \models \phi.$$

prf. of Claim: *By induction on the complexity of ϕ, unpack it into literals, positive literals will became a part of U. Only atomic sentences in U are valid in $\mathcal{A}$ by the Canonical model theorem.* □

A notion of consistency implies satisfiability cont.

Theorem

If N is a notion of consistency and $p \in N$, then p has a model.

Proof (cont.)

We have constructed $\bigcup \overline{p}$.
Let $U \subseteq \bigcup \overline{p}$ be the subset of all atomic sentences. By the tasks corresponding to (1) *and* (2) *we have that U is $=$-closed and so by the Canonical model theorem we have $\mathcal{A} \models U$.*
Claim: *For every L-sentence ϕ we have:*

$$\phi \in \bigcup \overline{p} \Rightarrow \mathcal{A} \models \phi.$$

prf. of Claim: *By induction on the complexity of ϕ, unpack it into literals, positive literals will became a part of U. Only atomic sentences in U are valid in $\mathcal{A}$ by the Canonical model theorem.* □
Since $p \subseteq \bigcup \overline{p}$, then $\mathcal{A} \models p$. □

Compactness theorem

Theorem (Compactness)

Let L be a language of cardinality λ. Let T be finitely satisfiable L-theory. Then T is satisfiable.

Compactness theorem

Theorem (Compactness)

Let L be a language of cardinality λ. Let T be finitely satisfiable L-theory. Then T is satisfiable.

Proof

Let $W := \{c_i; i < \lambda\}$ be a set of new constants. And let N be a set of sets of $L(W)$-sentences such that

$$p \in N$$

$$\Updownarrow$$

(i) Fewer than λ constants from W occur in sentences of p and

(ii) p is finitely satisfiable.

Compactness theorem

Theorem (Compactness)

Let L be a language of cardinality λ. Let T be finitely satisfiable L-theory. Then T is satisfiable.

Proof

Let $W := \{c_i; i < \lambda\}$ be a set of new constants. And let N be a set of sets of $L(W)$-sentences such that

$$p \in N$$
$$\Updownarrow$$

(i) Fewer than λ constants from W occur in sentences of p and

(ii) p is finitely satisfiable.

Compactness theorem

Theorem (Compactness)

Let L be a language of cardinality λ. Let T be finitely satisfiable L-theory. Then T is satisfiable.

Proof

Let $W := \{c_i; i < \lambda\}$ be a set of new constants. And let N be a set of sets of $L(W)$-sentences such that

$$p \in N$$
$$\Updownarrow$$

(i) Fewer than λ constants from W occur in sentences of p and

(ii) p is finitely satisfiable.

Notice that $T \in N$ since it contains no constant from W and is finitely satisfiable.

Compactness theorem

Theorem (Compactness)

Let L be a language of cardinality λ. Let T be finitely satisfiable L-theory. Then T is satisfiable.

Proof

Let $W := \{c_i; i < \lambda\}$ be a set of new constants. And let N be a set of sets of $L(W)$-sentences such that

$$p \in N$$

$$\Updownarrow$$

(i) Fewer than λ constants from W occur in sentences of p and

(ii) p is finitely satisfiable.

Notice that $T \in N$ since it contains no constant from W and is finitely satisfiable. It is easy to verify that N is a notion of consistency for $L(W)$ which by previous theorem implies that T is satisfiable! □

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.
- There were λ tasks to be performed in construction of $(p_i)_{i<\lambda}$ and each of the tasks could be completed, provided that λ many sets p_{i+1} were assigned to it.

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.
- There were λ tasks to be performed in construction of $(p_i)_{i<\lambda}$ and each of the tasks could be completed, provided that λ many sets p_{i+1} were assigned to it.
- Now image that each task τ belongs to some model theorist τ^*.

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.
- There were λ tasks to be performed in construction of $(p_i)_{i<\lambda}$ and each of the tasks could be completed, provided that λ many sets p_{i+1} were assigned to it.
- Now image that each task τ belongs to some model theorist τ^*.
- Every model theorist τ^* can think of himself as playing a game against all other model theorists:

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.
- There were λ tasks to be performed in construction of $(p_i)_{i<\lambda}$ and each of the tasks could be completed, provided that λ many sets p_{i+1} were assigned to it.
- Now image that each task τ belongs to some model theorist τ^*.
- Every model theorist τ^* can think of himself as playing a game against all other model theorists:

Notions of consistency and games

- Recall the theorem about satisfiability of conditions of notion of consistency.
- There were λ tasks to be performed in construction of $(p_i)_{i<\lambda}$ and each of the tasks could be completed, provided that λ many sets p_{i+1} were assigned to it.
- Now image that each task τ belongs to some model theorist τ^*.
- Every model theorist τ^* can think of himself as playing a game against all other model theorists: he wins the game is the task τ has been completed by the time the chain $(p_i)_{i<\lambda}$ is finished.
- **Then each model theorist has a winning strategy of his own game, provided that he can pick λ of the sets p_{i+1}.**

Games

Formal games

- A few words about what exactly we mean by games.

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).
- The players play the game by choosing one by one a sequence of objects $(x_i)_{i<\gamma}$

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).
- The players play the game by choosing one by one a sequence of objects $(x_i)_{i<\gamma}$
- Player $\forall$ decides x_i if $i \in M_\forall$.

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).
- The players play the game by choosing one by one a sequence of objects $(x_i)_{i<\gamma}$
- Player $\forall$ decides x_i if $i \in M_\forall$.
- Player $\exists$ decides x_i if $i \in M_\exists$.

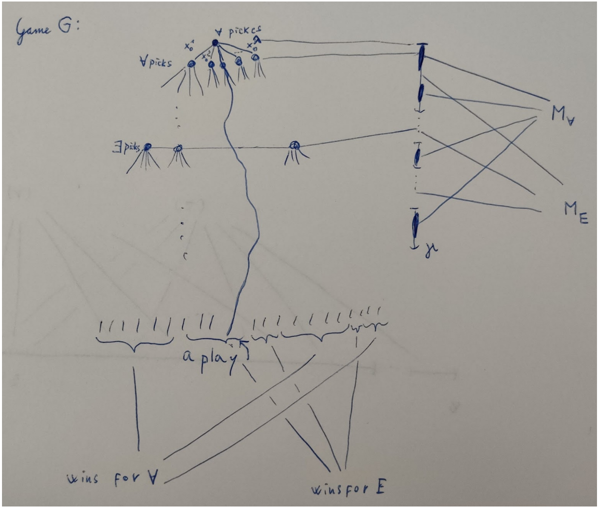
Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).
- The players play the game by choosing one by one a sequence of objects $(x_i)_{i<\gamma}$
- Player $\forall$ decides x_i if $i \in M_\forall$.
- Player $\exists$ decides x_i if $i \in M_\exists$.
- When the game is finished the result is a sequence $(x_i)_{i<\gamma}$ called play.

Formal games

- A few words about what exactly we mean by games.
- Always two players: $\forall$ (Abelard), $\exists$ (Eloise)
- Each game G has an ordinal length γ.
- γ is partitioned into two sets $M_\exists$ and $M_\forall$ (sets of moves for each player).
- The players play the game by choosing one by one a sequence of objects $(x_i)_{i<\gamma}$
- Player $\forall$ decides x_i if $i \in M_\forall$.
- Player $\exists$ decides x_i if $i \in M_\exists$.
- When the game is finished the result is a sequence $(x_i)_{i<\gamma}$ called play.
- The set of all plays in partitioned into **wins for** $\forall$ and **wins for** $\exists$.

A picture of a game



Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.
- A strategy σ for a player in a game G is called **winning** if the player wins every time they use σ.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.
- A strategy σ for a player in a game G is called **winning** if the player wins every time they use σ.
- A game G is **determined** if at least one player has a winning strategy.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.
- A strategy σ for a player in a game G is called **winning** if the player wins every time they use σ.
- A game G is **determined** if at least one player has a winning strategy.
- Let $\bar{y}$ be a position in a game G. **A strategy from $\bar{y}$ onwards** it is a set of instruction of how to play the game after the position $\bar{y}$ has been reached. Such partial strategy is **winning** if the player using it always wins.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.
- A strategy σ for a player in a game G is called **winning** if the player wins every time they use σ.
- A game G is **determined** if at least one player has a winning strategy.
- Let $\bar{y}$ be a position in a game G. **A strategy from $\bar{y}$ onwards** it is a set of instruction of how to play the game after the position $\bar{y}$ has been reached. Such partial strategy is **winning** if the player using it always wins.

Strategies

- Suppose that the players play the game up to but not including the β-th move. Then $(x_i)_{i<\beta}$ is called a **position** of length β. (Think path from the root of the tree of length β.)
- A **strategy** is a set of instruction on how to navigate the game tree.
- E.g. a **strategy** for $\exists$ player is a family of functions that is defined for positions of length $i \in M_\exists$.
- A strategy σ for a player in a game G is called **winning** if the player wins every time they use σ.
- A game G is **determined** if at least one player has a winning strategy.
- Let $\bar{y}$ be a position in a game G. **A strategy from $\bar{y}$ onwards** it is a set of instruction of how to play the game after the position $\bar{y}$ has been reached. Such partial strategy is **winning** if the player using it always wins. (For the player to use the strategy, $\bar{y}$ must have been reached.)
- A position $\bar{y}$ is called **winning** for a player if that player has a winning strategy from $\bar{y}$ onwards.

Exercise and game of finite length

Theorem

Let $\bar{y}$ is a position in a game G before the play has been finished. Then:

Exercise and game of finite length

Theorem

Let $\overline{y}$ is a position in a game G before the play has been finished. Then:

- *If $\overline{y}$ is winning for a player, then the player can guarantee to be still in a winning position after the next move.*

Exercise and game of finite length

Theorem

Let $\overline{y}$ is a position in a game G before the play has been finished. Then:

- *If $\overline{y}$ is winning for a player, then the player can guarantee to be still in a winning position after the next move.*
- *If $\overline{y}$ is not winning for a player, then the other player can guarantee that the former player is still not in the winning position after the next move.*

Exercise and game of finite length

Theorem

Let $\overline{y}$ is a position in a game G before the play has been finished. Then:

- *If $\overline{y}$ is winning for a player, then the player can guarantee to be still in a winning position after the next move.*
- *If $\overline{y}$ is not winning for a player, then the other player can guarantee that the former player is still not in the winning position after the next move.*

Exercise and game of finite length

Theorem

Let $\overline{y}$ is a position in a game G before the play has been finished. Then:

- *If $\overline{y}$ is winning for a player, then the player can guarantee to be still in a winning position after the next move.*
- *If $\overline{y}$ is not winning for a player, then the other player can guarantee that the former player is still not in the winning position after the next move.*

This implies that games of finite length are determined!

Forcing with games (a teaser)

Forcing with games

- So far, for a condition in a notion of consistency, we can built its model.

Forcing with games

- So far, for a condition in a notion of consistency, we can built its model.
- We will define what is a **notion of forcing**, which will be a special kind of **notions of consistency**.

Forcing with games

- So far, for a condition in a notion of consistency, we can built its model.
- We will define what is a **notion of forcing**, which will be a special kind of **notions of consistency**.
- We will want to construct models of a whole ascending chain of condition in the notion forcing.

Forcing with games

- So far, for a condition in a notion of consistency, we can built its model.
- We will define what is a **notion of forcing**, which will be a special kind of **notions of consistency**.
- We will want to construct models of a whole ascending chain of condition in the notion forcing.
- By noticing, that notions of forcing look like trees, we can play games on them.

Forcing with games

- So far, for a condition in a notion of consistency, we can built its model.
- We will define what is a **notion of forcing**, which will be a special kind of **notions of consistency**.
- We will want to construct models of a whole ascending chain of condition in the notion forcing.
- By noticing, that notions of forcing look like trees, we can play games on them.
- Studying these games, we will be able to understand properties of models which we will get from notions of forcing.

A picture of the situation.