

# Vingean reflection: Reliable reasoning for self-improving agents

Benja Fallenstein

Machine Intelligence Research Institute

May 16, 2015

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?
- Want solid **theoretical understanding** of problem & solution
  - What is correct reasoning and decision making?



## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?
- Want solid **theoretical understanding** of problem & solution
  - What is correct reasoning and decision making?
  - Probability theory, decision theory, game theory, statistical learning theory, Bayesian networks, formal verification, . . .

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?
- Want solid **theoretical understanding** of problem & solution
  - What is correct reasoning and decision making?
  - Probability theory, decision theory, game theory, statistical learning theory, Bayesian networks, formal verification, . . .
  - . . . go in the right direction, but *are not enough*.

## Motivation

- Smarter-than-human intelligence isn't around the corner
  - but it'll (probably) be developed eventually.
- Important to ensure it's **aligned with our interests**
  - But how do we *specify beneficial goals*?
  - How do we make sure system **actually pursues them**?
  - How do we *correct* the system if we get it wrong?
- Want solid **theoretical understanding** of problem & solution
  - What is correct reasoning and decision making?
  - Probability theory, decision theory, game theory, statistical learning theory, Bayesian networks, formal verification, . . .
  - . . . go in the right direction, but *are not enough*.
  - Need for **foundational research**—which can be done today.

## Vingean reflection

- Can we create a **self-modifying** system...
  - ... that goes through a **billion modifications**...
  - ... *without ever going wrong?*

## Vingean reflection

- Can we create a **self-modifying** system. . .
  - . . . that goes through a **billion modifications**. . .
  - . . . *without ever going wrong?*
  - Need *extremely reliable* way for an AI to reason about agents **smarter than itself** — much more reliable than a human!

## Vingean reflection

- Can we create a **self-modifying** system. . .
  - . . . that goes through a **billion modifications**. . .
  - . . . *without ever going wrong?*
  - Need *extremely reliable* way for an AI to reason about agents **smarter than itself** — much more reliable than a human!
- Need to use *abstract reasoning*
  - Vingean: Can't know exactly what a smarter successor will do
  - Instead, have *abstract* reasons to think its choices are good

## Vingean reflection

- Can we create a **self-modifying** system. . .
  - . . . that goes through a **billion modifications**. . .
  - . . . *without ever going wrong?*
  - Need *extremely reliable* way for an AI to reason about agents **smarter than itself** — much more reliable than a human!
- Need to use *abstract reasoning*
  - Vingean: Can't know exactly what a smarter successor will do
  - Instead, have *abstract* reasons to think its choices are good
  - Standard decision theory doesn't model this

## Vingean reflection

- Can we create a **self-modifying** system. . .
  - . . . that goes through a **billion modifications**. . .
  - . . . *without ever going wrong?*
  - Need *extremely reliable* way for an AI to reason about agents **smarter than itself** — much more reliable than a human!
- Need to use *abstract reasoning*
  - Vingean: Can't know exactly what a smarter successor will do
  - Instead, have *abstract* reasons to think its choices are good
  - Standard decision theory doesn't model this
- Formal logic as a model of abstract reasoning



- 1 The “procrastination paradox”
- 2 A formal toy model
- 3 Partial solutions
- 4 Logical uncertainty
- 5 Conclusions

## The “procrastination paradox”

- Agent in a deterministic, known world; discrete timesteps.
- In each timestep, the agent chooses whether to press a button:
  - If pressed in 1<sup>st</sup> round: Utility =  $1/2$
  - If pressed in 2<sup>nd</sup> round (and not before): Utility =  $2/3$
  - If pressed in 3<sup>rd</sup> round (and not before): Utility =  $3/4$
  - ...

## The “procrastination paradox”

- Agent in a deterministic, known world; discrete timesteps.
- In each timestep, the agent chooses whether to press a button:
  - If pressed in 1<sup>st</sup> round: Utility =  $1/2$
  - If pressed in 2<sup>nd</sup> round (and not before): Utility =  $2/3$
  - If pressed in 3<sup>rd</sup> round (and not before): Utility =  $3/4$
  - ...
  - If never pressed: Utility = 0

## The “procrastination paradox”

- Agent in a deterministic, known world; discrete timesteps.
- In each timestep, the agent chooses whether to press a button:
  - If pressed in 1<sup>st</sup> round: Utility =  $1/2$
  - If pressed in 2<sup>nd</sup> round (and not before): Utility =  $2/3$
  - If pressed in 3<sup>rd</sup> round (and not before): Utility =  $3/4$
  - ...
  - If never pressed: Utility = 0
- (No optimal strategy, but sure can beat 0!)

## The “procrastination paradox”

- Agent in a deterministic, known world; discrete timesteps.
- In each timestep, the agent chooses whether to press a button:
  - If pressed in 1<sup>st</sup> round: Utility =  $1/2$
  - If pressed in 2<sup>nd</sup> round (and not before): Utility =  $2/3$
  - If pressed in 3<sup>rd</sup> round (and not before): Utility =  $3/4$
  - ...
  - If never pressed: Utility = 0
- (No optimal strategy, but sure can beat 0!)
- The agent is programmed to press the button immediately...
  - ... *unless* it finds a “good argument” that the button will get pressed *later*.

## The agent reasons:

- Suppose I don’t press the button now.
- Either I press the button in the next step, or I don’t.

## The agent reasons:

- Suppose I don’t press the button now.
- Either I press the button in the next step, or I don’t.
  - If I *do*, the button gets pressed, good.

## The agent reasons:

- Suppose I don't press the button now.
- Either I press the button in the next step, or I don't.
  - If I *do*, the button gets pressed, good.
  - If I *don't*, I must have found a good argument that the button gets pressed later. So the button gets pressed, good!



## The agent reasons:

- Suppose I don't press the button now.
- Either I press the button in the next step, or I don't.
  - If I *do*, the button gets pressed, good.
  - If I *don't*, I must have found a good argument that the button gets pressed later. So the button gets pressed, good!
  - Either way, the button gets pressed.

## The agent reasons:

- Suppose I don't press the button now.
- Either I press the button in the next step, or I don't.
  - If I *do*, the button gets pressed, good.
  - If I *don't*, I must have found a good argument that the button gets pressed later. So the button gets pressed, good!
  - Either way, the button gets pressed.

So the agent can always find a “good argument” that the button will get pressed later. . .

- . . . and therefore never presses the button!

## The agent reasons:

- Suppose I don’t press the button now.
- Either I press the button in the next step, or I don’t.
  - If I *do*, the button gets pressed, good.
  - If I *don’t*, I must have found a good argument that the button gets pressed later. So the button gets pressed, good!
  - Either way, the button gets pressed.

So the agent can always find a “good argument” that the button will get pressed later. . .

- . . . and therefore never presses the button!

*If we want to have **reliable self-referential reasoning**, we must understand how to **avoid this paradox** (and others like it).*

## So what went wrong? (And how do we fix it?)

## So what went wrong? (And how do we fix it?)

- The paradox doesn't go through with finite time horizons—
  - —or with temporal discounting:
  - Utility =  $\sum_{t=0}^{\infty} \gamma_t \cdot R_t$ , where  $\sum_{t=0}^{\infty} \gamma_t < \infty$  and  $R_t \in [0, 1]$ .

## So what went wrong? (And how do we fix it?)

- The paradox doesn't go through with finite time horizons—
  - —or with temporal discounting:
  - Utility =  $\sum_{t=0}^{\infty} \gamma_t \cdot R_t$ , where  $\sum_{t=0}^{\infty} \gamma_t < \infty$  and  $R_t \in [0, 1]$ .
- Does using temporal discounting fix all such problems?

## So what went wrong? (And how do we fix it?)

- The paradox doesn't go through with finite time horizons—
  - —or with temporal discounting:
  - Utility =  $\sum_{t=0}^{\infty} \gamma_t \cdot R_t$ , where  $\sum_{t=0}^{\infty} \gamma_t < \infty$  and  $R_t \in [0, 1]$ .
- Does using temporal discounting fix all such problems?
- In our toy model:
  - **No**, not by itself.
    - Still get (more technical) paradoxes of self-reference.

## So what went wrong? (And how do we fix it?)

- The paradox doesn't go through with finite time horizons—
  - —or with temporal discounting:
  - Utility =  $\sum_{t=0}^{\infty} \gamma_t \cdot R_t$ , where  $\sum_{t=0}^{\infty} \gamma_t < \infty$  and  $R_t \in [0, 1]$ .
- Does using temporal discounting fix all such problems?
- In our toy model:
  - **No**, not by itself.
    - Still get (more technical) paradoxes of self-reference.
  - But: there are ways to fix these problems. . .
  - . . . which work **if** we use finite horizons or discounting.
    - (Suggests this is key to avoiding the problem.)



- 1 The “procrastination paradox”
- 2 A formal toy model
- 3 Partial solutions
- 4 Logical uncertainty
- 5 Conclusions


- **For our toy model**, use formal logic.
- But *not* because we think realistic smarter-than-human agents work like this.
  - The **problem** seems to be much more general.
  - Any scheme for highly reliable self-referential reasoning will need to deal with it somehow.

- **For our toy model**, use formal logic.
- But *not* because we think realistic smarter-than-human agents work like this.
  - The **problem** seems to be much more general.
  - Any scheme for highly reliable self-referential reasoning will need to deal with it somehow.
- Rather: because we can prove theorems about it—
  - and then see what this tells us about the real problem.

- Write  $P(n)$  for “the button is pressed in the  $n^{\text{th}}$  timestep”.
- Define computable function  $f(n)$ :
  - $f(n)$  searches for proofs
    - in Peano Arithmetic (PA)
    - of length  $\leq 10^{100+n}$
    - of “ $\exists k > n. P(k)$ ” — i.e., “button pressed later”.
  - If proof found  $\implies$  returns 0 (“don’t press button”).
  - Else  $\implies$  returns 1 (“press button”).
- $\text{PA} \vdash P(n) \leftrightarrow [f(n) = 1]$ .
  - (Self-referential definition by Kleene’s second recursion thm.)


- By looking at  $f(n + 1)$ , can prove (in  $\ll 10^{100+n}$  symbols):

---

<sup>1</sup>**Notation:**  $\Box_{\text{PA}} \ulcorner \varphi \urcorner$  means " $\varphi$  is provable in PA". 


- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - "Either the button will be pressed in the next timestep or not":  
 $PA \vdash P(n+1) \vee \neg P(n+1)$

---

<sup>1</sup>Notation:  $\Box_{PA} \ulcorner \varphi \urcorner$  means " $\varphi$  is provable in PA". 


- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - "Either the button will be pressed in the next timestep or not":  
 $PA \vdash P(n+1) \vee \neg P(n+1)$
  - "If button not pressed in next step, must have found proof it will be pressed later":<sup>1</sup>  
 $PA \vdash \neg P(n+1) \rightarrow \Box_{PA} \lceil \exists k > n+1. P(k) \rceil$

---

<sup>1</sup>**Notation:**  $\Box_{PA} \lceil \varphi \rceil$  means " $\varphi$  is provable in PA". 

- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - *"Either the button will be pressed in the next timestep or not":*  
 $PA \vdash P(n+1) \vee \neg P(n+1)$
  - *"If button not pressed in next step, must have found proof it will be pressed later":<sup>1</sup>*  
 $PA \vdash \neg P(n+1) \rightarrow \Box_{PA} \lceil \exists k > n+1. P(k) \rceil$
  - **(???)** *"If there's a proof that the button will be pressed, then it will indeed be pressed."*  
 $PA \vdash \Box_{PA} \lceil \exists k > n+1. P(k) \rceil \rightarrow \exists k > n+1. P(k)$

---

<sup>1</sup>**Notation:**  $\Box_{PA} \lceil \varphi \rceil$  means " $\varphi$  is provable in PA". 




- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - "Either the button will be pressed in the next timestep or not":  
 $PA \vdash P(n+1) \vee \neg P(n+1)$
  - "If button not pressed in next step, must have found proof it will be pressed later":<sup>1</sup>  
 $PA \vdash \neg P(n+1) \rightarrow \Box_{PA} \lceil \exists k > n+1. P(k) \rceil$
  - (???) "If there's a proof that the button will be pressed, then it will indeed be pressed."  
 $PA \vdash \Box_{PA} \lceil \exists k > n+1. P(k) \rceil \rightarrow \exists k > n+1. P(k)$
  - "Hence, either way, the button is pressed."  
 $PA \vdash P(n+1) \vee \exists k > n+1. P(k)$   
 $PA \vdash \exists k > n. P(k)$

<sup>1</sup>Notation:  $\Box_{PA} \lceil \varphi \rceil$  means " $\varphi$  is provable in PA".

- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - "Either the button will be pressed in the next timestep or not":  
 $PA \vdash P(n+1) \vee \neg P(n+1)$
  - "If button not pressed in next step, must have found proof it will be pressed later":<sup>1</sup>  
 $PA \vdash \neg P(n+1) \rightarrow \Box_{PA} \lceil \exists k > n+1. P(k) \rceil$
  - (???) "If there's a proof that the button will be pressed, then it will indeed be pressed."  
 $PA \vdash \Box_{PA} \lceil \exists k > n+1. P(k) \rceil \rightarrow \exists k > n+1. P(k)$
  - "Hence, either way, the button is pressed."  
 $PA \vdash P(n+1) \vee \exists k > n+1. P(k)$   
 $PA \vdash \exists k > n. P(k)$
- Hence,  $f(n) = 0$  (for all  $n \in \mathbb{N}$ )... button never pressed.

<sup>1</sup>Notation:  $\Box_{PA} \lceil \varphi \rceil$  means " $\varphi$  is provable in PA".

- By looking at  $f(n+1)$ , can prove (in  $\ll 10^{100+n}$  symbols):
  - "Either the button will be pressed in the next timestep or not":  
 $PA \vdash P(n+1) \vee \neg P(n+1)$
  - "If button not pressed in next step, must have found proof it will be pressed later":<sup>1</sup>  
 $PA \vdash \neg P(n+1) \rightarrow \Box_{PA} \lceil \exists k > n+1. P(k) \rceil$
  - (???) "If there's a proof that the button will be pressed, then it will indeed be pressed."  
 $PA \vdash \Box_{PA} \lceil \exists k > n+1. P(k) \rceil \rightarrow \exists k > n+1. P(k)$
  - "Hence, either way, the button is pressed."  
 $PA \vdash P(n+1) \vee \exists k > n+1. P(k)$   
 $PA \vdash \exists k > n. P(k)$
- Hence,  $f(n) = 0$  (for all  $n \in \mathbb{N}$ )... button never pressed.
- $\implies$  So  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .

<sup>1</sup>Notation:  $\Box_{PA} \lceil \varphi \rceil$  means " $\varphi$  is provable in PA". 

- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?

- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?
- Why do we think our agent will work correctly?
  - We reason: "It will take only actions if it has very good reason to believe these actions will be safe —"

- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?
- Why do we think our agent will work correctly?
  - We reason: “It will take only actions if it has very good reason to believe these actions will be safe — therefore, any actions it will take will be almost certainly safe.”

- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?
- Why do we think our agent will work correctly?
  - We reason: “It will take only actions if it has very good reason to believe these actions will be safe — therefore, any actions it will take will be almost certainly safe.”
  - An agent should be able to use the same argument when reasoning about rewriting itself!

- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?
- Why do we think our agent will work correctly?
  - We reason: “It will take only actions if it has very good reason to believe these actions will be safe — therefore, any actions it will take will be almost certainly safe.”
  - An agent should be able to use the same argument when reasoning about rewriting itself!
- Need *something* like  $T \vdash \Box_T \lceil \varphi \rceil \rightarrow \varphi \dots$



- PA avoids the paradox since  $PA \not\vdash \Box_{PA} \lceil \varphi \rceil \rightarrow \varphi$ .
  - $\rightarrow$  Generalize this beyond our logic-based toy example?
- Why do we think our agent will work correctly?
  - We reason: "It will take only actions if it has very good reason to believe these actions will be safe — therefore, any actions it will take will be almost certainly safe."
  - An agent should be able to use the same argument when reasoning about rewriting itself!
- Need *something* like  $T \vdash \Box_T \lceil \varphi \rceil \rightarrow \varphi \dots$ 
  - Gödel/Löb: But that's inconsistent, finite time horizons or not!

- 1 The “procrastination paradox”
- 2 A formal toy model
- 3 Partial solutions**
- 4 Logical uncertainty
- 5 Conclusions

## Partial solutions

- ① Can have theories  $T_0, T_1, T_2, \dots$  s.t.  $T_{n+1} \vdash \Box_{T_n} \lceil \varphi \rceil \rightarrow \varphi$ .
- Agent using  $T_{n+1}$  can rewrite into agent using  $T_n$ .
  - Stops working when we reach  $T_0$ .
  - Works for finite time horizons.

## Partial solutions

- 1 Can have theories  $T_0, T_1, T_2, \dots$  s.t.  $T_{n+1} \vdash \Box_{T_n} \lceil \varphi \rceil \rightarrow \varphi$ .
  - Agent using  $T_{n+1}$  can rewrite into agent using  $T_n$ .
  - Stops working when we reach  $T_0$ .
  - Works for finite time horizons.
- 2 Can have theories s.t.  $T_n \vdash \Box_{T_{n+1}} \lceil \varphi \rceil \rightarrow \varphi$  for all  $\varphi \in \Pi_1$ .
  - Agent using  $T_n$  can rewrite into agent using  $T_{n+1}$ .
  - Can rewrite forever!
    - (**But:** Agent doesn't know this! :-())
  - Works with temporal discounting (Fallenstein & Soares, 2014).

## Partial solutions

- 1 Can have theories  $T_0, T_1, T_2, \dots$  s.t.  $T_{n+1} \vdash \Box_{T_n} \lceil \varphi \rceil \rightarrow \varphi$ .
  - Agent using  $T_{n+1}$  can rewrite into agent using  $T_n$ .
  - Stops working when we reach  $T_0$ .
  - Works for finite time horizons.
- 2 Can have theories s.t.  $T_n \vdash \Box_{T_{n+1}} \lceil \varphi \rceil \rightarrow \varphi$  for all  $\varphi \in \Pi_1$ .
  - Agent using  $T_n$  can rewrite into agent using  $T_{n+1}$ .
  - Can rewrite forever!
    - (**But:** Agent doesn't know this! :-())
  - Works with temporal discounting (Fallenstein & Soares, 2014).

Do these approaches generalize beyond formal logic?

- 1 The “procrastination paradox”
- 2 A formal toy model
- 3 Partial solutions
- 4 Logical uncertainty**
- 5 Conclusions

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!



## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!
- Example: Choose between  $O(n^2)$  and  $O(n \log n)$  algorithm

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!
- Example: Choose between  $O(n^2)$  and  $O(n \log n)$  algorithm
- Realistic Vingean reflection needs logical uncertainty.

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!
- Example: Choose between  $O(n^2)$  and  $O(n \log n)$  algorithm
- Realistic Vingean reflection needs logical uncertainty.
- Approach for study:
  - Probability distribution over *complete theories* in some first-order language.

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!
- Example: Choose between  $O(n^2)$  and  $O(n \log n)$  algorithm
- Realistic Vingean reflection needs logical uncertainty.
- Approach for study:
  - Probability distribution over *complete theories* in some first-order language.
  - e.g. complete theories extending Peano Arithmetic (PA)
    - $\rightarrow$  uncertainty about whether PA is consistent

## Logical uncertainty

- Standard probability theory = *environmental* uncertainty.
  - Agents are assumed to be *logically omniscient*.
  - No theoretical understanding of mathematical uncertainty!
- Example: Choose between  $O(n^2)$  and  $O(n \log n)$  algorithm
- Realistic Vingean reflection needs logical uncertainty.
- Approach for study:
  - Probability distribution over *complete theories* in some first-order language.
  - e.g. complete theories extending Peano Arithmetic (PA)
    - $\rightarrow$  uncertainty about whether PA is consistent
  - Reflection is still difficult

## Reflection in probabilistic logic

- Assign probabilities  $\mathbb{P}[\varphi]$  to sentences  $\varphi \dots$ 
  - ... in a language with a symbol for  $\mathbb{P}[\cdot]$ .
  - Require e.g.: if  $\text{ZFC} \vdash \varphi \rightarrow \psi$ , then  $\mathbb{P}[\varphi] \leq \mathbb{P}[\psi]$ .

## Reflection in probabilistic logic

- Assign probabilities  $\mathbb{P}[\varphi]$  to sentences  $\varphi \dots$ 
  - $\dots$  in a language with a symbol for  $\mathbb{P}[\cdot]$ .
  - Require e.g.: if  $\text{ZFC} \vdash \varphi \rightarrow \psi$ , then  $\mathbb{P}[\varphi] \leq \mathbb{P}[\psi]$ .
- Reflection:  $\alpha \leq \mathbb{P}[\varphi] \leq \beta \implies \mathbb{P}[\alpha \leq \mathbb{P}[\varphi] \leq \beta] = 1$ .

## Reflection in probabilistic logic

- Assign probabilities  $\mathbb{P}[\varphi]$  to sentences  $\varphi \dots$ 
  - $\dots$  in a language with a symbol for  $\mathbb{P}[\cdot]$ .
  - Require e.g.: if  $\text{ZFC} \vdash \varphi \rightarrow \psi$ , then  $\mathbb{P}[\varphi] \leq \mathbb{P}[\psi]$ .
- Reflection:  $\alpha \leq \mathbb{P}[\varphi] \leq \beta \implies \mathbb{P}[\alpha \leq \mathbb{P}[\varphi] \leq \beta] = 1$ .
  - But let  $\text{ZFC} \vdash \varphi \leftrightarrow \mathbb{P}[\varphi] < 1$  (diagonal lemma).
  - Suppose  $\mathbb{P}[\varphi] = 1$ . Then  $\mathbb{P}[\varphi] = \mathbb{P}[\mathbb{P}[\varphi] < 1] = 0$ .



## Reflection in probabilistic logic

- Assign probabilities  $\mathbb{P}[\varphi]$  to sentences  $\varphi \dots$ 
  - ... in a language with a symbol for  $\mathbb{P}[\cdot]$ .
  - Require e.g.: if  $\text{ZFC} \vdash \varphi \rightarrow \psi$ , then  $\mathbb{P}[\varphi] \leq \mathbb{P}[\psi]$ .
- Reflection:  $\alpha \leq \mathbb{P}[\varphi] \leq \beta \implies \mathbb{P}[\alpha \leq \mathbb{P}[\varphi] \leq \beta] = 1$ .
  - But let  $\text{ZFC} \vdash \varphi \leftrightarrow \mathbb{P}[\varphi] < 1$  (diagonal lemma).
  - Suppose  $\mathbb{P}[\varphi] = 1$ . Then  $\mathbb{P}[\varphi] = \mathbb{P}[\mathbb{P}[\varphi] < 1] = 0$ .
  - Suppose  $\mathbb{P}[\varphi] \leq 1 - \varepsilon < 1$ . Then  
 $\mathbb{P}[\varphi] = \mathbb{P}[\mathbb{P}[\varphi] < 1] \geq \mathbb{P}[\mathbb{P}[\varphi] \leq 1 - \varepsilon] = 1$ .
  - Contradiction!

## Reflection in probabilistic logic

- Assign probabilities  $\mathbb{P}[\varphi]$  to sentences  $\varphi \dots$ 
  - ... in a language with a symbol for  $\mathbb{P}[\cdot]$ .
  - Require e.g.: if  $\text{ZFC} \vdash \varphi \rightarrow \psi$ , then  $\mathbb{P}[\varphi] \leq \mathbb{P}[\psi]$ .
- Reflection:  $\alpha \leq \mathbb{P}[\varphi] \leq \beta \implies \mathbb{P}[\alpha \leq \mathbb{P}[\varphi] \leq \beta] = 1$ .
  - But let  $\text{ZFC} \vdash \varphi \leftrightarrow \mathbb{P}[\varphi] < 1$  (diagonal lemma).
  - Suppose  $\mathbb{P}[\varphi] = 1$ . Then  $\mathbb{P}[\varphi] = \mathbb{P}[\mathbb{P}[\varphi] < 1] = 0$ .
  - Suppose  $\mathbb{P}[\varphi] \leq 1 - \varepsilon < 1$ . Then  
 $\mathbb{P}[\varphi] = \mathbb{P}[\mathbb{P}[\varphi] < 1] \geq \mathbb{P}[\mathbb{P}[\varphi] \leq 1 - \varepsilon] = 1$ .
  - Contradiction!
- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1$

## Procrastination in probabilistic logic

- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1.$
- Let  $ZFC \vdash P(n) \leftrightarrow \mathbb{P}[\exists k > n. P(k)] < 1 - \frac{1}{n}$ 
  - “Button pressed in step  $n$  unless very sure it’s pressed later”

## Procrastination in probabilistic logic

- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1.$
- Let  $\text{ZFC} \vdash P(n) \leftrightarrow \mathbb{P}[\exists k > n. P(k)] < 1 - \frac{1}{n}$ 
  - “Button pressed in step  $n$  unless very sure it’s pressed later”
  - $\mathbb{P}[\exists n. P(n)] = 1$
  - For all  $n$ ,  $\mathbb{P}[P(n)] = 0$

## Procrastination in probabilistic logic

- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1$ .
- Let  $ZFC \vdash P(n) \leftrightarrow \mathbb{P}[\exists k > n. P(k)] < 1 - \frac{1}{n}$ 
  - "Button pressed in step  $n$  unless very sure it's pressed later"
  - $\mathbb{P}[\exists n. P(n)] = 1$
  - For all  $n$ ,  $\mathbb{P}[P(n)] = 0$
- Unclear how to interpret this!
  - $\mathbb{P}$  can't be  $\sigma$ -additive probability measure on standard models

## Procrastination in probabilistic logic

- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1$ .
- Let  $ZFC \vdash P(n) \leftrightarrow \mathbb{P}[\exists k > n. P(k)] < 1 - \frac{1}{n}$ 
  - “Button pressed in step  $n$  unless very sure it’s pressed later”
  - $\mathbb{P}[\exists n. P(n)] = 1$
  - For all  $n$ ,  $\mathbb{P}[P(n)] = 0$
- Unclear how to interpret this!
  - $\mathbb{P}$  can’t be  $\sigma$ -additive probability measure on standard models
  - But can be finitely additive measure

## Procrastination in probabilistic logic

- Christiano (2013): consistent to have for all  $\alpha, \beta \in \mathbb{Q}$ , all  $\varphi$ :  
 $\alpha < \mathbb{P}[\varphi] < \beta \implies \mathbb{P}[\alpha < \mathbb{P}[\varphi] < \beta] = 1$ .
- Let  $\text{ZFC} \vdash P(n) \leftrightarrow \mathbb{P}[\exists k > n. P(k)] < 1 - \frac{1}{n}$ 
  - “Button pressed in step  $n$  unless very sure it’s pressed later”
  - $\mathbb{P}[\exists n. P(n)] = 1$
  - For all  $n$ ,  $\mathbb{P}[P(n)] = 0$
- Unclear how to interpret this!
  - $\mathbb{P}$  can’t be  $\sigma$ -additive probability measure on standard models
  - But can be finitely additive measure
  - Clearer understanding needed!

- 1 The “procrastination paradox”
- 2 A formal toy model
- 3 Partial solutions
- 4 Logical uncertainty
- 5 Conclusions**



## Conclusions

- Gave example of self-referential reasoning gone wrong.
  - Any **reliable** system for self-referential reasoning will need to deal with this somehow.
- **Analyzed** the problem using a **toy model**,
  - and looked for solutions that generalize.
  - Can extend to utility-based agents (Fallenstein & Soares, 2014)

## Conclusions

- Gave example of self-referential reasoning gone wrong.
  - Any **reliable** system for self-referential reasoning will need to deal with this somehow.
- **Analyzed** the problem using a **toy model**,
  - and looked for solutions that generalize.
  - Can extend to utility-based agents (Fallenstein & Soares, 2014)
- Looked for extensions to logical uncertainty.
  - Reflection is still difficult.
  - Still get versions of the procrastination paradox.
  - Better understanding needed.

## Conclusions

- Gave example of self-referential reasoning gone wrong.
  - Any **reliable** system for self-referential reasoning will need to deal with this somehow.
- **Analyzed** the problem using a **toy model**,
  - and looked for solutions that generalize.
  - Can extend to utility-based agents (Fallenstein & Soares, 2014)
- Looked for extensions to logical uncertainty.
  - Reflection is still difficult.
  - Still get versions of the procrastination paradox.
  - Better understanding needed.
- Extremely reliable self-referential reasoning isn't trivial...
  - but we can make progress towards it! **Thanks for listening!**