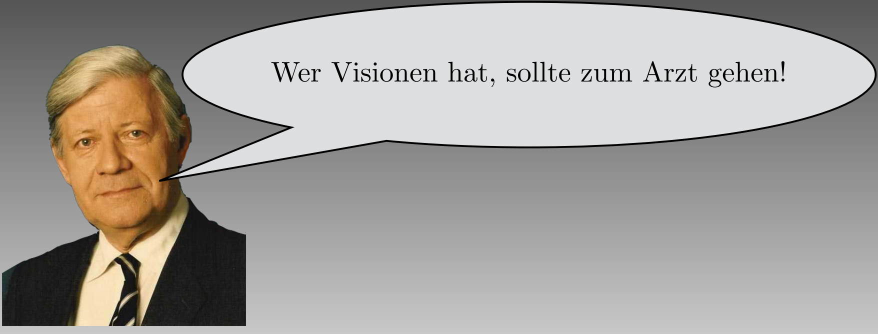
Deep Reasoning

A Vision for Automated Deduction

Stephan Schulz

Deep Reasoning

A Vision for Automated Deduction



Deep Reasoning

A Vision for Automated Deduction

Anybody with visions should go see a doctor!

Agenda

- Introduction
- Deep Learning
- Automated Theorem Proving
- Deep Reasoning
- Conclusion

Introduction: Historical Perspective

1955	Logic Theorist
1956	Dartmouth Workshop - “Birth of AI”
1957	Perceptron
1958	LISP
1960	Davis-Putnam (DPLL 1962)
1965	Resolution/Unification
1970	Knuth-Bendix Completion
1972	PROLOG (1983 WAM)
1965-1975	MLP/back propagation
1980s	Expert systems/Planners
1986	Decision tree learning
1990-1994	Superposition calculus
since 1997	Development of E (E 0.3 January 1999)
since ca. 2005	“Deep Learning”
2008	E 1.0

Deep Learning

Deep Learning - Introduction

- Instance of machine learning
- Typical setting: Supervised learning
  - Large number of pre-classified examples
  - Examples are presented with expected output
  - System learns classification/evaluation
- Result: Trained model
  - Will provide classification/evaluation when presented with new input

Deep Learning - Methods

- Application of known techniques on a new scale
  - Supervised learning (classification/evaluation/association)
  - Artificial neural networks
  - Gradient-based learning/back-propagation
- New:
  - Big networks
  - Complex network structure
    - Multiple sub-networks
    - Convolution layers
    - Recurrence
  - (Mostly) raw input
    - Feature extraction is part of the learning
    - Encoding is part of the learning

Deep Learning - Successes

- AI used to have problems with "easy" tasks
- Deep learning successfully addresses these problems
  - Image recognition
  - Voice recognition
  - Natural language translation
  - Hard games
    - Video games (real time)
    - Go
    - Poker

AlphaGo

Deep Learning - Successes

- AI used to have problems with "easy" tasks
- Deep learning successfully addresses these problems
  - Image recognition
  - Voice recognition
  - Natural language translation
  - Hard games
    - Video games (real time)
    - Go
    - Poker

AlphaGo

Deep learning drives resurgence of Artificial Intelligence!

Deep Learning - Why Now?

- Popularity of Deep Learning
  - ... slowly growing since the mid 2000s
  - ... explosively growing since mid 2010s
- Driven by “big hardware”
  - Clusters of computers
  - ... with clusters of GPUs
- Driven by “big data”
  - Large training sets
  - Large size of individuals
- Driven by Open Source
  - Algorithms and models published under permissive licenses
  - Many state-of-the-art machine learning libraries available

Deep Learning - A Parable

Cast of Characters

Deep Learning - A Parable

Cast of Characters

Neanderthal Man

Cast of Characters

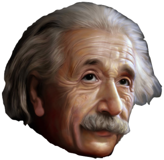
Neanderthal Man

Sir Isaac Newton

Cast of Characters

Neanderthal Man

Sir Isaac Newton



Dr. Albert Einstein

Neanderthal Learning

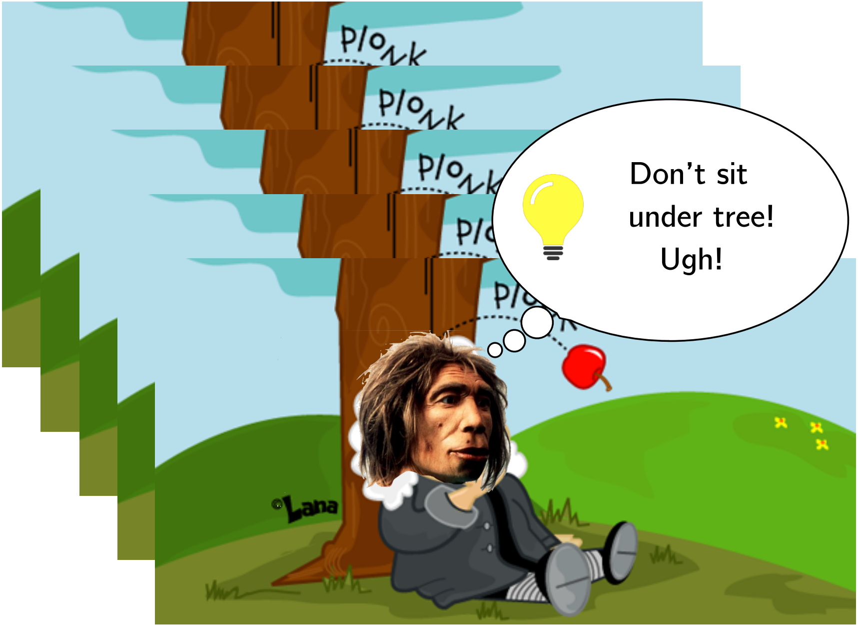
Neanderthal Learning

Neanderthal Learning

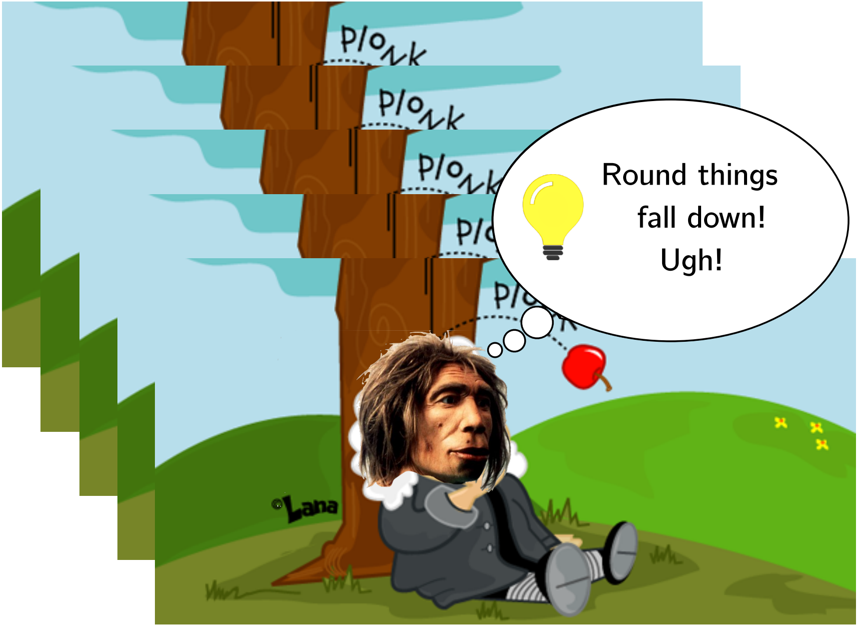
Neanderthal Learning

Neanderthal Learning

Neanderthal Learning



Neanderthal Learning



Enlightenment!

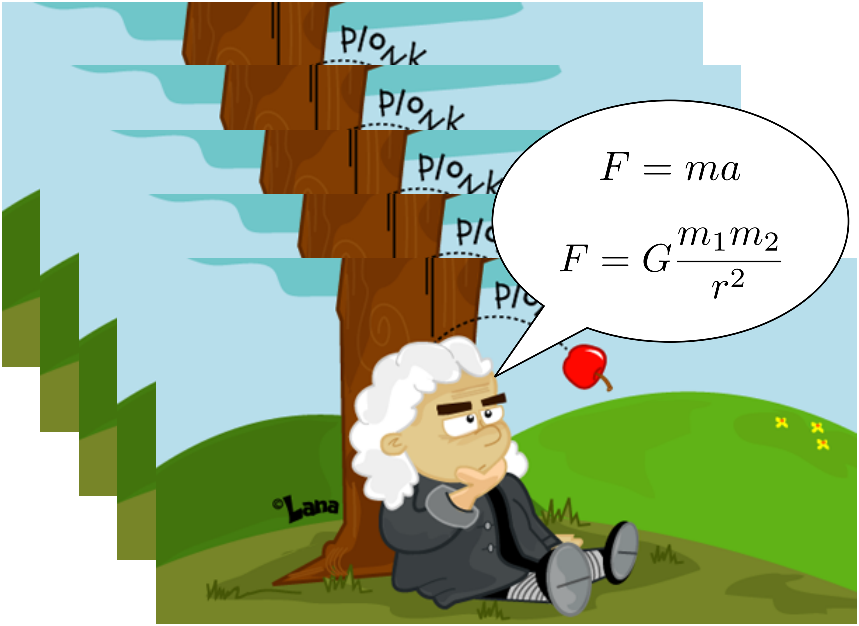
Enlightenment!

Enlightenment!

Enlightenment!

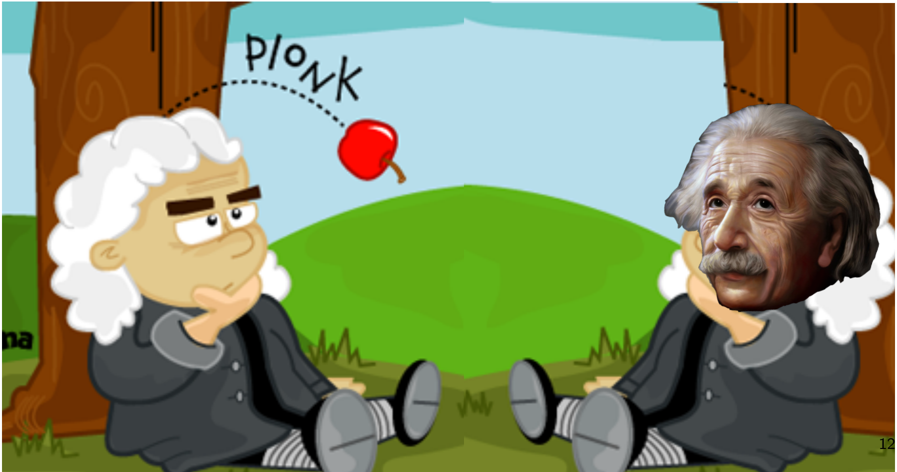
Enlightenment!

Enlightenment!




PlONK

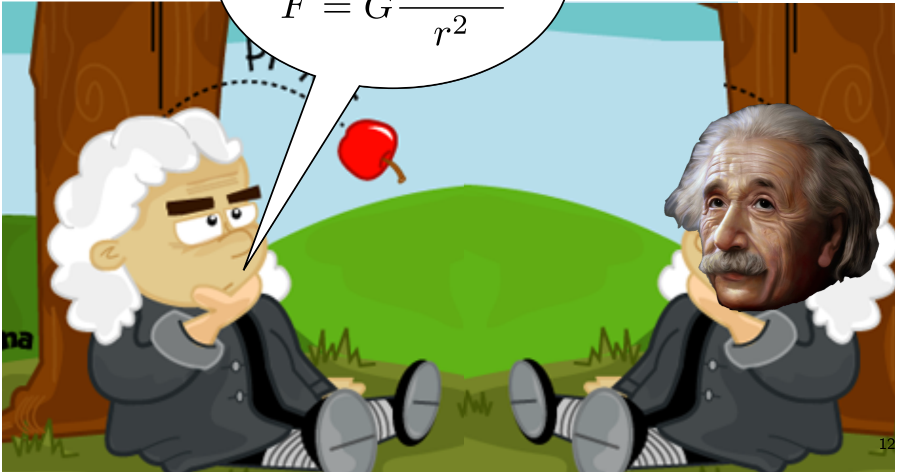
Compare and Contrast



Compare and Contrast

$$F = ma$$

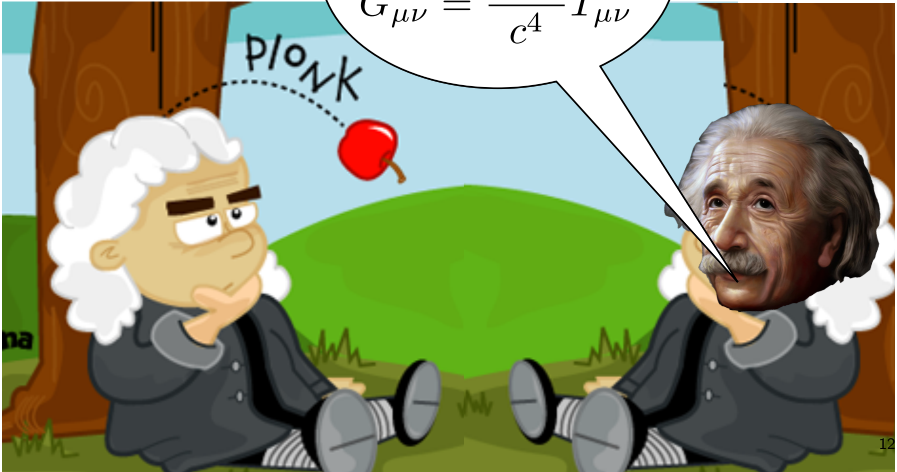
$$F = G\frac{m_1 m_2}{r^2}$$



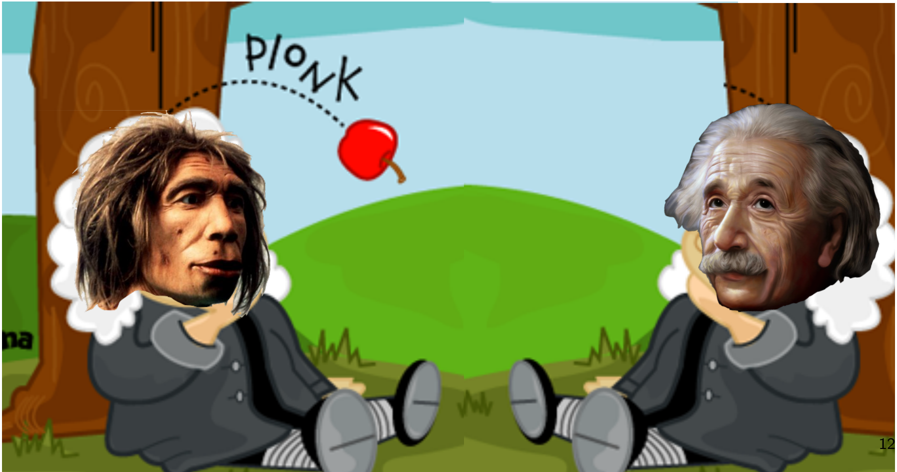
Compare and Contrast

$$E = mc^2$$

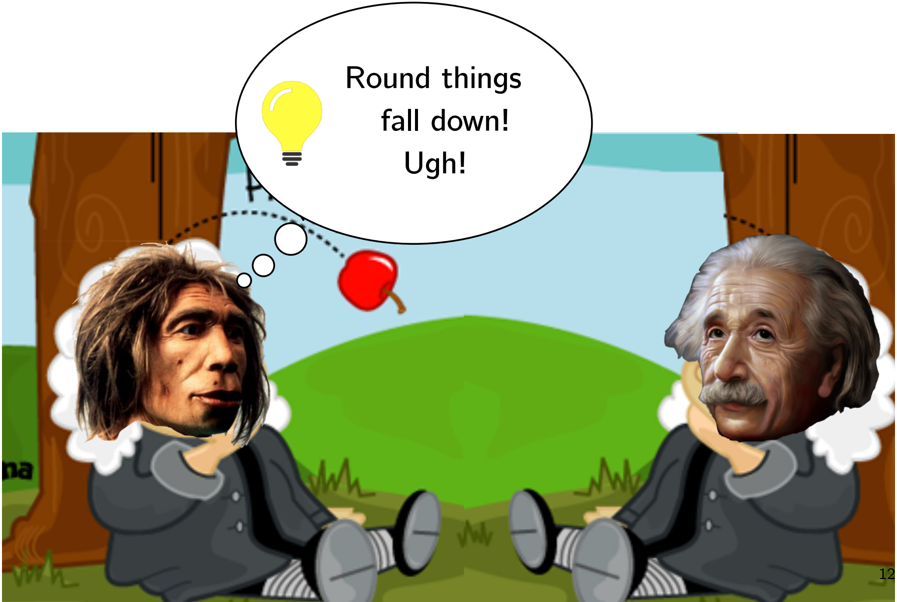
$$G_{\mu\nu} = \frac{8\pi G}{c^4} T_{\mu\nu}$$



Compare and Contrast



Compare and Contrast



Compare and Contrast

Deep Learning Weaknesses

- Computationally expensive
  - Big models use specialized hardware for training
  - Even model application has non-trivial cost
- Knowledge is represented by large set distributed weights
  - Low inherent level of abstraction
  - Model is noisy
- Knowledge is largely inaccessible
  - Hard to understand
  - Hard to explain
  - Hard to communicate

Deep Learning Weaknesses

- Computationally expensive
  - Big models use specialized hardware for training
  - Even model application has non-trivial cost
- Knowledge is represented by large set distributed weights
  - Low inherent level of abstraction
  - Model is noisy
- Knowledge is largely inaccessible
  - Hard to understand
  - Hard to explain
  - Hard to communicate

Unsupported claim (still true):
Deep learning alone will run into natural limits!

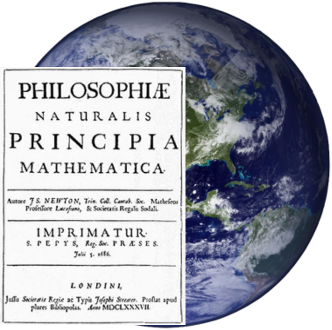
Automated Theorem Proving

Theorem Proving: Big Picture

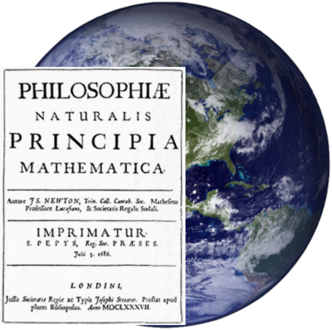
Real World Problem

Theorem Proving: Big Picture

Real World Problem



Real World Problem


PHILOSOPHIÆ
NATURALIS
PRINCIPIA
MATHEMATICA

Autore J S. NEWTON, Trin. Coll. Cantab. Soc. Matheseos Professore Lucasiano, & Societatis Regalis Sodali.

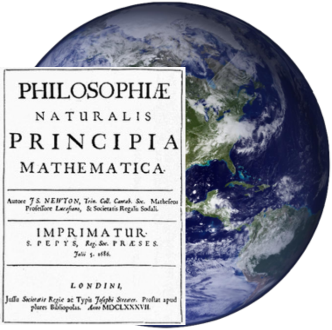
IMPRIMATUR.
S. PEPYS, Reg. Soc. PRÆSES.
Julii 5. 1686.

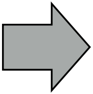
LONDINI,
Jussu Societatis Regiæ ac Typis Josephi Streater. Prostat apud plures Bibliopolas. Anno MDCLXXXVII.

Formalized Problem

Theorem Proving: Big Picture

Real World Problem





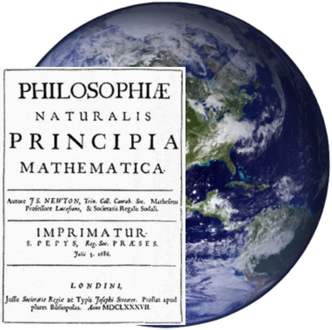
Formalized Problem

$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

$$\overset{?}{\models}$$

$$mortal(socrates)$$

Real World Problem



Formalized Problem

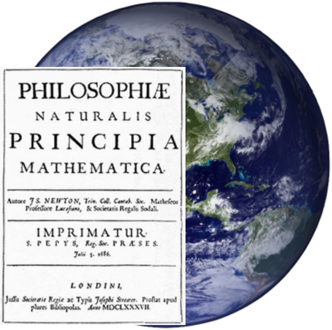
$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

$$\overset{?}{\models}$$

$$mortal(socrates)$$

Theorem Proving: Big Picture

Real World Problem



Formalized Problem

$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

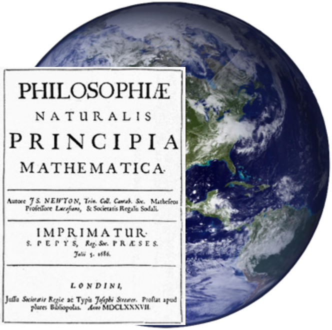
$$\overset{?}{\models}$$

$$mortal(socrates)$$

ATP

Theorem Proving: Big Picture

Real World Problem



Formalized Problem

$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

$$\stackrel{?}{\models}$$

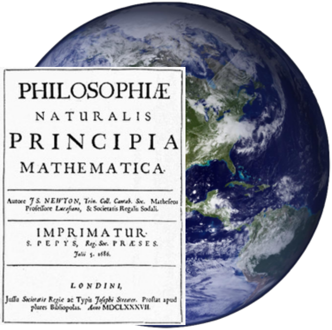
$$mortal(socrates)$$

ATP

Proof

Theorem Proving: Big Picture

Real World Problem



Formalized Problem

$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

$$\overset{?}{\models}$$

$$mortal(socrates)$$

ATP

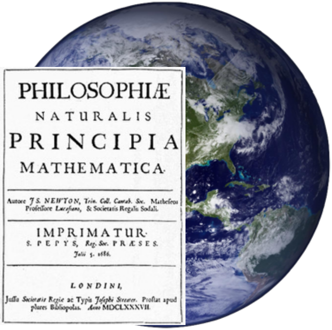
Proof

or

Countermodel

Theorem Proving: Big Picture

Real World Problem



Formalized Problem

$$\forall X : human(X) \rightarrow mortal(X)$$
$$\forall X : philosopher(X) \rightarrow human(X)$$
$$philosopher(socrates)$$

$$\overset{?}{\models}$$

$$mortal(socrates)$$

ATP

Proof

or

Countermodel

or

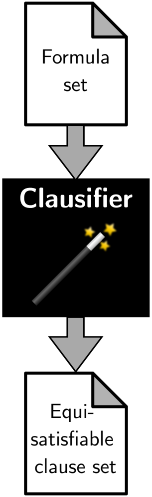
Timeout

Logics of Interest

- Propositional logic
  - SAT-solving: relatively independent sub-field
- First-order logics
  - ...with free symbols
  - ...with free symbols and equality
  - ...with background theories
  - ...with free symbols and background theories
- Higher order logics
  - Currently developing field

Contradiction and Saturation

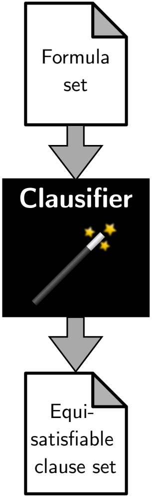
- Proof by contradiction
  - Assume negation of conjecture
  - Show that axioms and negated conjecture imply falsity
- Saturation
  - Convert problem to Clause Normal Form
  - Systematically enumerate logical consequences of axioms and negated conjecture
  - Goal: Explicit contradiction (empty clause)
- Redundancy elimination
  - Use contracting inferences to simplify or eliminate some clauses



Contradiction and Saturation

- Proof by contradiction
  - Assume negation of conjecture
  - Show that axioms and negated conjecture imply falsity
- Saturation
  - Convert problem to Clause Normal Form
  - Systematically enumerate logical consequences of axioms and negated conjecture
  - Goal: Explicit contradiction (empty clause)
- Redundancy elimination
  - Use contracting inferences to simplify or eliminate some clauses

Search control problem: How and in which order do we enumerate consequences?



Proof Search

```
# SZS status Theorem
# SZS output start CNFRefutation
fof(pel55_4, axiom, (![X1]:![X2]:(killed(X1,X2)=>hates(X1,X2))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/P
fof(pel55_1, axiom, (?[X1]:(lives(X1)&killed(X1,agatha))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/PUZ001+
fof(pel55_3, axiom, (![X1]:(lives(X1)=>((X1=agatha|X1=butler)|X1=charles))), file('/Users/schulz/EPROVER/TPTP_
fof(pel55_10, axiom, (![X1]:?[X2]:~(hates(X1,X2))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/PUZ001+1.p', p
fof(pel55_9, axiom, (![X1]:(hates(agatha,X1)=>hates(butler,X1))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/
fof(pel55_5, axiom, (![X1]:![X2]:(killed(X1,X2)=>~(richer(X1,X2)))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FL
fof(pel55_8, axiom, (![X1]:(~(richer(X1,agatha))=>hates(butler,X1))), file('/Users/schulz/EPROVER/TPTP_6.4.0_F
fof(pel55_6, axiom, (![X1]:(hates(agatha,X1)=>~(hates(charles,X1)))), file('/Users/schulz/EPROVER/TPTP_6.4.0_F
fof(pel55_7, axiom, (![X1]:(X1!=butler=>hates(agatha,X1))), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/PUZ001
fof(pel55_11, axiom, (agatha!=butler), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/PUZ001+1.p', pel55_11)).
fof(pel55, conjecture, (killed(agatha,agatha)), file('/Users/schulz/EPROVER/TPTP_6.4.0_FLAT/PUZ001+1.p', pel55
fof(c_0_11, plain, (![X3]:![X4]:(~killed(X3,X4)|hates(X3,X4))), inference(variable_rename,[status(thm)],[infer
fof(c_0_12, plain, ((lives(esk1_0)&killed(esk1_0,agatha))), inference(skolemize,[status(esa)],[inference(varia
fof(c_0_13, plain, (![X2]:(~lives(X2)|((X2=agatha|X2=butler)|X2=charles))), inference(variable_rename,[status(
cnf(c_0_14,plain,(hates(X1,X2)|~killed(X1,X2)), inference(split_conjunct,[status(thm)],[c_0_11])).
cnf(c_0_15,plain,(killed(esk1_0,agatha)), inference(split_conjunct,[status(thm)],[c_0_12])).
cnf(c_0_16,plain,(X1=charles|X1=butler|X1=agatha|~lives(X1)), inference(split_conjunct,[status(thm)],[c_0_13])
cnf(c_0_17,plain,(lives(esk1_0)), inference(split_conjunct,[status(thm)],[c_0_12])).
fof(c_0_18, plain, (![X3]:~hates(X3,esk2_1(X3))), inference(skolemize,[status(esa)],[inference(variable_rename
fof(c_0_19, plain, (![X2]:(~hates(agatha,X2)|hates(butler,X2))), inference(variable_rename,[status(thm)],[infe
fof(c_0_20, plain, (![X3]:![X4]:(~killed(X3,X4)|~richer(X3,X4))), inference(variable_rename,[status(thm)],[inf
fof(c_0_21, plain, (![X2]:(richer(X2,agatha)|hates(butler,X2))), inference(variable_rename,[status(thm)],[infe
fof(c_0_22, plain, (![X2]:(~hates(agatha,X2)|~hates(charles,X2))), inference(variable_rename,[status(thm)],[in
cnf(c_0_23,plain,(hates(esk1_0,agatha)), inference(spm,[status(thm)],[c_0_14, c_0_15])).
cnf(c_0_24,plain,(esk1_0=charles|esk1_0=butler|esk1_0=agatha), inference(spm,[status(thm)],[c_0_16, c_0_17])).
cnf(c_0_25,plain,(~hates(X1,esk2_1(X1))), inference(split_conjunct,[status(thm)],[c_0_18])).
cnf(c_0_26,plain,(hates(butler,X1)|~hates(agatha,X1)), inference(split_conjunct,[status(thm)],[c_0_19])).
fof(c_0_27, plain, (![X2]:(X2=butler|hates(agatha,X2))), inference(variable_rename,[status(thm)],[inference(fo
cnf(c_0_28,plain,(~richer(X1,X2)|~killed(X1,X2)), inference(split_conjunct,[status(thm)],[c_0_20])).
cnf(c_0_29,plain,(hates(butler,X1)|richer(X1,agatha)), inference(split_conjunct,[status(thm)],[c_0_21])).
cnf(c_0_30,plain,(~hates(charles,X1)|~hates(agatha,X1)), inference(split_conjunct,[status(thm)],[c_0_22])).
cnf(c_0_31,plain,(esk1_0=agatha|esk1_0=butler|hates(charles,agatha)), inference(spm,[status(thm)],[c_0_23, c_0
```

Proof Search

```
# SZS output start CNFRefutation
fof(pel55_4, axiom, (![X1]:![X2]:(killed(X1,X2)=>hates(X1,X2))),
    file('PUZ001+1.p', pel55_4)).
...
fof(pel55, conjecture, (killed(agatha,agatha)),
    file('PUZ001+1.p', pel55)).
...
fof(c_0_12, plain, ((lives(esk1_0)&killed(esk1_0,agatha))),
    inference(skolemize,[status(esa)],
    [inference(variable_rename,[status(thm)],[pel55_1])])).
...
cnf(c_0_14,plain,(hates(X1,X2)|~killed(X1,X2)),
    inference(split_conjunct,[status(thm)],[c_0_11])).
...
cnf(c_0_23,plain,(hates(esk1_0,agatha)),
    inference(spm,[status(thm)],[c_0_14, c_0_15])).
...
cnf(c_0_45,plain,($false),
    inference(sr,[status(thm)],[inference(rw,[status(thm)],
    [c_0_15, c_0_43]), c_0_44]), ['proof']).
# SZS output end CNFRefutation
```

Proof Search and Choice Points

- First-order logic is semi-decidable
  - Provers search for proof in infinite space
  - . . . of possible derivations
  - . . . of possible consequences
- Major choice points of Superposition calculus:
  - Term ordering (which terms are bigger)
  - (Negative) literal selection
  - Selection of clauses for inferences (with the given clause algorithm)

Some Properties of ATP

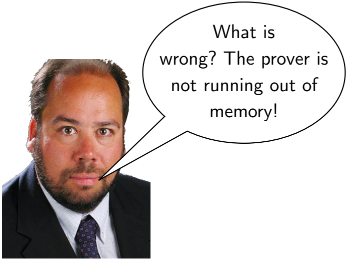
- Individual operations cheap(ish)
  - Computing one consequence is no problem
  - Computing 1000 consequences is no problem
- But: Large/infinite search space
  - 1000 consequences is usually enough for a proof
  - ... but rarely enough to find it!
- Combinatorial explosion
  - High branching factor
  - Simplification helps a lot
  - ... but not nearly enough!

Big Data and ATP

- Automated tuning of theorem provers since the 1990s
  - Examples:
    - E-SETHEO schedules
    - E's automatic auto mode
    - Vampire's *black magic* box
  - Based on performance only
- Reason: Proof search traces are big!
  - . . . really big!
  - . . . and theorem provers are memory-limited anyways

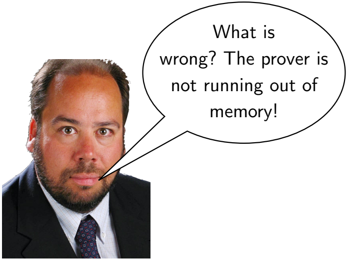
Big Data and ATP

- Automated tuning of theorem provers since the 1990s
  - Examples:
    - E-SETHEO schedules
    - E's automatic auto mode
    - Vampire's *black magic* box
  - Based on performance only
- Reason: Proof search traces are big!
  - ... really big!
  - ... and theorem provers are memory-limited anyways
- Ca. 2014: Something wonderful happens
  - Hardware finally catches up
  - Implementation techniques improve



Big Data and ATP

- Automated tuning of theorem provers since the 1990s
  - Examples:
    - E-SETHEO schedules
    - E's automatic auto mode
    - Vampire's *black magic* box
  - Based on performance only
- Reason: Proof search traces are big!
  - ... really big!
  - ... and theorem provers are memory-limited anyways
- Ca. 2014: Something wonderful happens
  - Hardware finally catches up
  - Implementation techniques improve



We can finally afford to look DEEPLY into proofs!

Deep Reasoning

Vision: Search Control

- Long-term goal: Extract search control knowledge
  - . . . from examples of successful proof searches
  - . . . from examples of failing proof searches
- Primary use case: Clause selection
  - Which of the current candidate consequences should be considered first?
  - Extract good/bad search decisions from proof protocols

Vision: Search Control

- Long-term goal: Extract search control knowledge
  - ... from examples of successful proof searches
  - ... from examples of failing proof searches
- Primary use case: Clause selection
  - Which of the current candidate consequences should be considered first?
  - Extract good/bad search decisions from proof protocols
- It's happening!
  - Premise selection (Urban, Irving, et al)
  - Clause Selection (Loos, Irvin, Kaliszyk et al) - see next session

Vision: Search Control

- Long-term goal: Extract search control knowledge
  - . . . from examples of successful proof searches
  - . . . from examples of failing proof searches
- Primary use case: Clause selection
  - Which of the current candidate consequences should be considered first?
  - Extract good/bad search decisions from proof protocols
- It's happening!
  - Premise selection (Urban, Irving, et al)
  - Clause Selection (Loos, Irvin, Kaliszyk et al) - see next session

Vision: Automated Scientist

- Setting: Background theory+examples
  - Background theory in explicit logic
  - Examples
- Process
  - Deep learner hypothesizes relationship
  - Hypothesis is converted to symbolic logic (*Magic happens here*)
  - ATP system checks hypotheses for consistency with background theory
    - Failure: Abduction can refine hypothesis
    - Success: Tentatively add hypothesis to theory
  - ATP system generates new consequences to test on examples

Vision: Fully Interactive AI

- Setting: Rational agent interacting with environment
- Deep learner:
  - Vision
  - Voice
  - Language
  - Suggest actions
- Symbolic reasoning system
  - Hard-coded world knowledge
  - Hard-coded constraints on behavior

The End

Conclusion

- Deep learning and symbolic reasoning are complementary
- Hardware is now finally sufficient for both
  - . . . even in combined systems
- We're looking forward to an interesting future

Conclusion

- Deep learning and symbolic reasoning are complementary
- Hardware is now finally sufficient for both
  - ... even in combined systems
- We're looking forward to an interesting future

And when the time comes to decide whether to switch on the new, improved AI that is vastly superior to humans and will eliminate all errors, a couple of imperial bureaucrats will gather round a table, and one will say: "We've already paid for it, so let's switch it on"...

Conclusion

- Deep learning and symbolic reasoning are complementary
- Hardware is now finally sufficient for both
  - ...even in combined systems
- We're looking forward to an interesting future

And when the time comes to decide whether to switch on the new, improved AI that is vastly superior to humans and will eliminate all errors, a couple of imperial bureaucrats will gather round a table, and one will say: "We've already paid for it, so let's switch it on"...

Marc Uwe Kling (as "the Kangaroo")

Thank you!
Questions? Discussion?