Mining Human Proofs from Machine Proofs

ENSPM 2021

Tuesday, 12 July 2021

Paulo Oliva
Queen Mary University of London
(joint work with Rob Arthan)

Machine Proof	Human Proof
Difficult to "understand"	Focus on understanding "why" theorem is true
All formulas treated the same way	More "interesting" sub-goals highlighted as "Lemmas"
Prove exactly what needs to be proven	Generalising often makes proofs easier/shorter
No new concepts	Special treatment for new "important" ideas/ constructions

Is A provable in L?

formula →

Is A provable in L?

formula → Is A provable in L? ← logic

formula

logic

Is A provable in L?

reduce

formula

logic

Is A provable in L?

reduce

Is E true in C?

formula

logic

Is A provable in L?

reduce

Is E true in C?

equation

formula

logic

Is A provable in L?

reduce

Is E true in C?

equation

class of algebras

formula → Is A provable in L? ← logic

reduce

Is E true in C?

equation → ↻ ← class of algebras

machine finds
equational proof

formula

logic

Is A provable in L?

reduce

recover natural deduction proof

Is E true in C?

equation

class of algebras

machine finds equational proof

Is

$$\neg\neg(\neg\neg\neg A \to A)$$

provable in intuitionistic Lukasiewicz logic?

Does

$$\neg\neg(\neg\neg\neg x \to x) = 0$$

hold in all bounded hoops?

pocrim

$\langle X, \otimes, \to, \geq, 0 \rangle$

partially ordered
commutative
residuated integral
monoid

hoops

Buchi/Owens'74

pocrims satisfying:

If $x \geq y$ then

$$x = y \otimes (y \to x)$$

prover9

- Automated theorem prover for first-order and equational logic
- Successor of Otter
- Developed by Bill McCune
- Uses resolution and paramodulation

http://www.cs.unm.edu/~mccune/prover9/

- Able to find proofs using prover9

- Able to find proofs using prover9
- Initially, not much out of the proofs other than that the result was true

```
40 x + (x ==> 1) = 1.  [copy(39),flip(a)].
41 1 ==> x = y ==> ((y ==> 1) ==> x).  [para(40(a,1),5(a,1,1))].
42 x ==> ((x ==> 1) ==> y) = 1 ==> y.  [copy(41),flip(a)].
43 x + 1 = y + (x + (y ==> 1)).  [para(40(a,1),18(a,1,2))].
44 1 = y + (x + (y ==> 1)).  [para(9(a,1),43(a,1))].
45 x + (y + (x ==> 1)) = 1.  [copy(44),flip(a)].
46 x + (y ==> (x ==> z)) = (y ==> z) + ((y ==> z) ==> x).  [para(22(a,1),6(a,1,2))].
47 (x ==> y) + ((x ==> y) ==> z) = z + (x ==> (z ==> y)).  [copy(46),flip(a)].
48 x ==> 0 = y ==> (x ==> y).  [para(7(a,1),22(a,1,2))].
49 0 = y ==> (x ==> y).  [para(8(a,1),48(a,1))].
50 x ==> (y ==> x) = 0.  [copy(49),flip(a)].
51 x ==> 0 = y ==> (x ==> ((y ==> z) ==> z)).  [para(29(a,1),22(a,1,2))].
52 0 = y ==> (x ==> ((y ==> z) ==> z)).  [para(8(a,1),51(a,1))].
53 x ==> (y ==> ((x ==> z) ==> z)) = 0.  [copy(52),flip(a)].
54 1 ==> x = 0.  [para(37(a,1),7(a,1))].
55 x ==> ((x ==> 1) ==> y) = 0.  [para(54(a,1),42(a,2))].
56 1 = x + ((x ==> y) + (y ==> 1)).  [para(45(a,1),24(a,1))].
57 x + ((x ==> y) + (y ==> 1)) = 1.  [copy(56),flip(a)].
58 x ==> (0 ==> y) = (z ==> x) ==> (((z ==> x) ==> x) ==> y).  [para(50(a,1),26(a,1,2,1))].
59 x ==> y = (z ==> x) ==> (((z ==> x) ==> x) ==> y).  [para(33(a,1),58(a,1,2))].
60 (x ==> y) ==> (((x ==> y) ==> y) ==> z) = y ==> z.  [copy(59),flip(a)].
61 x ==> ((x ==> y) ==> ((y ==> z) ==> z)) = 0.  [para(26(a,1),53(a,1))].
62 x + (0 + (((x ==> y) ==> y) ==> 1)) = 1.  [para(29(a,1),57(a,1,2,1))].
63 x + (((x ==> y) ==> y) ==> 1) = 1.  [para(20(a,1),62(a,1,2))].
64 x + ((y ==> z) + ((y ==> z) ==> u)) = u + (x + (y ==> (u ==> z))).  [para(22(a,1),38(a,2,2,2))].
65 1 ==> x = y ==> ((((y ==> z) ==> z) ==> 1) ==> x).  [para(63(a,1),5(a,1,1))].
66 0 = y ==> ((((y ==> z) ==> z) ==> 1) ==> x).  [para(54(a,1),65(a,1))].
67 x ==> ((((x ==> y) ==> y) ==> 1) ==> z) = 0.  [copy(66),flip(a)].
68 (x ==> y) + ((x ==> y) ==> (x ==> 1)) = (x ==> 1) + 0.  [para(55(a,1),47(a,2,2))].
69 (x ==> y) + (x ==> ((x ==> y) ==> 1)) = (x ==> 1) + 0.  [para(22(a,1),68(a,1,2))].
70 (x ==> y) + (x ==> ((x ==> y) ==> 1)) = 0 + (x ==> 1).  [para(3(a,1),69(a,2))].
71 (x ==> y) + (x ==> ((x ==> y) ==> 1)) = x ==> 1.  [para(20(a,1),70(a,2))].
```

- Able to find proofs using prover9
- Initially, not much out of the proofs other than that the result was true

- Able to find proofs using prover9
- Initially, not much out of the proofs other than that the result was true
- But we started noticing some patterns...

Certain derived connectives kept appearing:

weak conjunction

$$A \wedge B \equiv A \otimes (A \to B)$$

strong disjunction

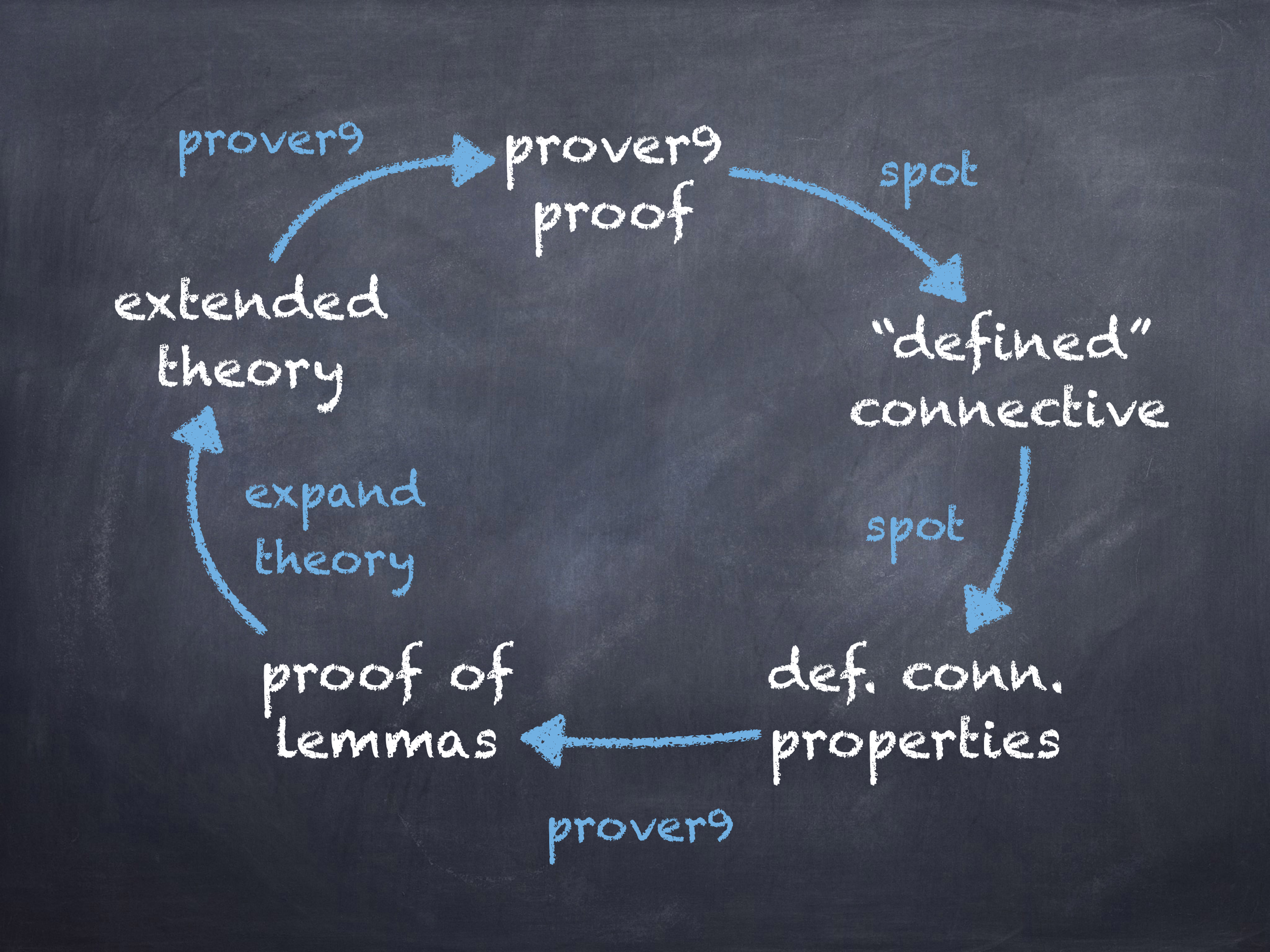
$$A \vee B \equiv (B \to A) \to A$$

strong implication

$$A \Rightarrow B \equiv A \to A \otimes B$$

NOR, Peirce's ampheck

$$A \downarrow B \equiv \neg A \otimes (B \to A)$$


prover9
prover9 proof
spot
extended theory
"defined" connective
expand theory
spot
proof of lemmas
def. conn. properties
prover9

Lemma 4.2 ($\mathbf{\text{ŁL}_i}$) $A \otimes B \;\leftrightarrow\; A \otimes (B \vee (A \Rightarrow B))$

Theorem 4.7 ($\mathbf{\text{ŁL}_i}$) $B \downarrow A \;\leftrightarrow\; A \downarrow B$

Corollary 4.8 ($\mathbf{\text{ŁL}_i}$) $(A^{\perp\perp} \multimap A)^{\perp\perp}$

Proof: Note that, since $\perp \leftrightarrow\; A \otimes A^{\perp}$ we have $(*)$ $A^{\perp\perp} \;\leftrightarrow\; A^{\perp} \Rightarrow A$. Moreover, it is easy to check that $(**)$ $X \downarrow (Y \multimap X) \;\leftrightarrow\; X^{\perp} \otimes (X \vee Y)$, for all X and Y. Hence

$$
\begin{aligned}
(A^{\perp\perp} \multimap A)^{\perp} &\leftrightarrow ((A^{\perp} \Rightarrow A) \multimap A)^{\perp} && (*)\\
&\leftrightarrow ((A^{\perp} \Rightarrow A) \multimap A)^{\perp} \otimes \underline{(A \multimap ((A^{\perp} \Rightarrow A) \multimap A))} && ([\mathsf{WK}])\\
&\leftrightarrow ((A^{\perp} \Rightarrow A) \multimap A) \downarrow A && (\text{def } \downarrow)\\
&\leftrightarrow A \downarrow ((A^{\perp} \Rightarrow A) \multimap A) && (\text{Theorem } 4.7)\\
&\leftrightarrow A^{\perp} \otimes (A \vee (A^{\perp} \Rightarrow A)) && (**)\\
&\leftrightarrow A^{\perp} \otimes A && (\text{Lemma } 4.2)\\
&\leftrightarrow \perp . && \blacksquare
\end{aligned}
$$

Conclusions

Conclusions

- Successfully mined human-readable proofs from machine proofs

Conclusions

- Successfully mined human-readable proofs from machine proofs
- Human input is identifying the "right" abstractions:
  - Find useful derived concepts
  - Recover an intuitive proof plan

Conclusions

- Successfully mined human-readable proofs from machine proofs
- Human input is identifying the "right" abstractions:
  - Find useful derived concepts
  - Recover an intuitive proof plan
- Automated support for proof refactoring?

Conclusions

- Successfully mined human-readable proofs from machine proofs
- Human input is identifying the "right" abstractions:
  - Find useful derived concepts
  - Recover an intuitive proof plan
- Automated support for proof refactoring?
- AI to automate human aspect?

Conclusions

- Successfully mined human-readable proofs from machine proofs
- Human input is identifying the "right" abstractions:
  - Find useful derived concepts
  - Recover an intuitive proof plan
- Automated support for proof refactoring?
- AI to automate human aspect?
- The late Bill McCune is the real star!