

Proof Attempts Cooperating via Models

Giles Reger

With Dmitry Tishkovsky and Andrei Voronkov

University of Manchester

15th September 2015

AVATAR: Guiding proof search using Models

- A new architecture implemented in Vampire
- Idea presented at CAV 14 and CADE 15
- Very high level idea:
 1. Represent the problem in a SAT solver
 2. Construct a Model
 3. If no model, return “unsat”
 4. Use the Model to select a sub-problem to explore
 5. If sub-problem is refuted, learn something and goto 1
 6. If strategy is complete return “sat”

Creating the SAT problem with Splitting

- The splitting basics:
 - For variable disjoint clauses C_1 and C_2
 - $S \cup (C_1 \vee C_2)$ is unsat iff both $S \cup C_1$ and $S \cup C_2$ are
 - Consider $S \cup C_1$ and $S \cup C_2$ separately
- For every clause C in the problem
 - Let $D_1 \vee \dots \vee D_n$ be its minimal variable-disjoint components
 - Consistently introduce a name p_i for component D_i
 - Add SAT clause $p_1 \vee \dots \vee p_n$
- A model of this SAT problem is a splitting decision

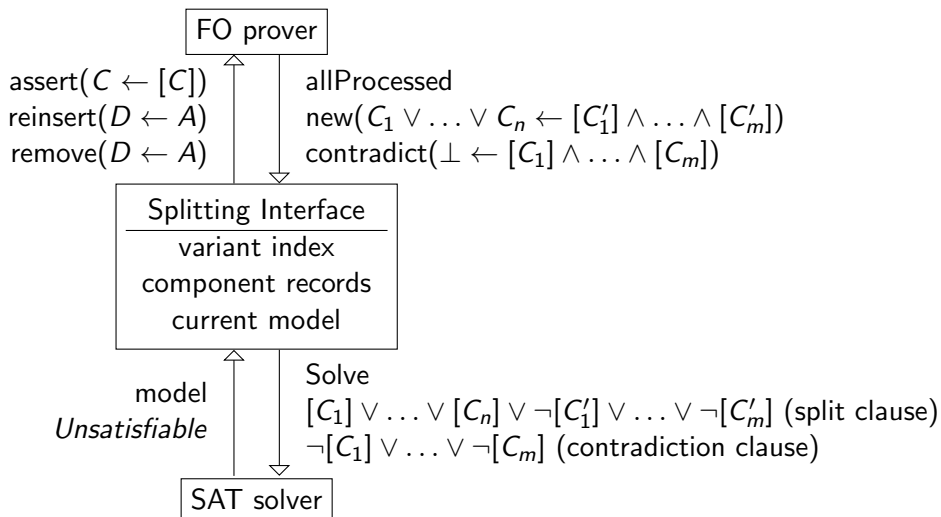
When refuting a model

- First-order reasoning tracks which parts of the model derived clauses depend on by labelling clauses
- A will depend on some part of the model
 - Derive $\perp \mid \{p_1, \dots, p_n\}$
 - Learn/add the conflict clause $\neg p_1 \vee \dots \vee \neg p_n$
 - Now reconstruct the model
- This represents backtracking
- The conflict clause blocks a family of possible models/splitting decisions

Missing details

- Process is incremental, set of clauses expanding
- When updating SAT model need to add/remove clauses from FO solver
- Simplifications may be conditional on current model as it can change
- Many variations i.e. may not add clause that cannot be split to SAT solver

AVATAR Architecture



Example

- Input:

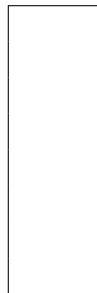
$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

SAT



Components



Example

- Input:

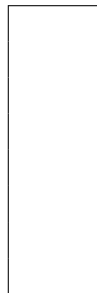
$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

SAT



Components



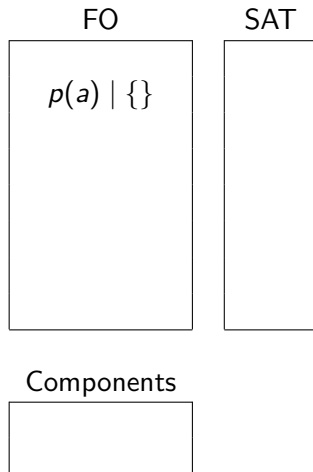
Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- **FO: Process new clauses**
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation



Example

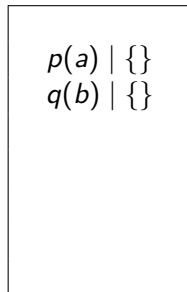
- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- **FO: Process new clauses**
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO



SAT



Components



Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a) \mid \{\}$
$q(b) \mid \{\}$

SAT

$$1 \vee 2$$

Components

$1 \mapsto \neg p(x)$
$2 \mapsto \neg q(y)$

Example

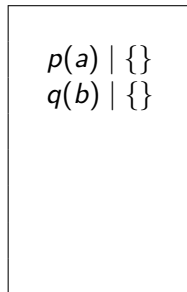
- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model**
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

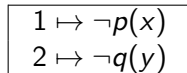
FO



SAT

$\underline{1} \vee 2$

Components



Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$		$\{\}$
$q(b)$		$\{\}$
$\neg p(x)$		$\{1\}$

SAT

$$\underline{1} \vee 2$$

Components

1	\mapsto	$\neg p(x)$
2	\mapsto	$\neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$		$\{\}$
$q(b)$		$\{\}$
$\neg p(x)$		$\{1\}$
\perp		$\{1\}$

SAT

$$\underline{1} \vee 2$$

Components

1	\mapsto	$\neg p(x)$
2	\mapsto	$\neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$	$ \ \{\}$
$q(b)$	$ \ \{\}$
$\neg p(x)$	$ \ \{1\}$
\perp	$ \ \{1\}$

SAT

$$\underline{1} \vee 2$$

$$\neg 1$$

Components

1	$\mapsto \neg p(x)$
2	$\mapsto \neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model**
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$	$ \ \{\}$
$q(b)$	$ \ \{\}$
$\neg p(x)$	$ \ \{1\}$
\perp	$ \ \{1\}$

SAT

$$1 \vee \underline{2}$$

$$\underline{\neg 1}$$

Components

1	$\mapsto \neg p(x)$
2	$\mapsto \neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$		$\{\}$
$q(b)$		$\{\}$
$\neg p(x)$		$\{1\}$
\perp		$\{1\}$
$\neg q(y)$		$\{2\}$

SAT

$$1 \vee \underline{2}$$

$$\underline{\neg 1}$$

Components

1	\mapsto	$\neg p(x)$
2	\mapsto	$\neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$		$\{\}$
$q(b)$		$\{\}$
$\neg p(x)$		$\{1\}$
\perp		$\{1\}$
$\neg q(y)$		$\{2\}$
\perp		$\{2\}$

SAT

$$1 \vee \underline{2}$$

$$\underline{\neg 1}$$

Components

1	\mapsto	$\neg p(x)$
2	\mapsto	$\neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$	$ \ \{\}$
$q(b)$	$ \ \{\}$
$\neg p(x)$	$ \ \{1\}$
\perp	$ \ \{1\}$
$\neg q(y)$	$ \ \{2\}$
\perp	$ \ \{2\}$

SAT

$$1 \vee \underline{2}$$

$$\underline{\neg 1}$$

$$\neg 2$$

Components

1	$\mapsto \neg p(x)$
2	$\mapsto \neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model**
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a)$		$\{\}$
$q(b)$		$\{\}$
$\neg p(x)$		$\{1\}$
\perp		$\{1\}$
$\neg q(y)$		$\{2\}$
\perp		$\{2\}$

SAT

$$1 \vee 2$$

$$\neg 1$$

$$\neg 2$$

Components

1	\mapsto	$\neg p(x)$
2	\mapsto	$\neg q(y)$

Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

- Refutation

- From the SAT solver

FO

$p(a)$	$ \ \{\}$
$q(b)$	$ \ \{\}$
$\neg p(x)$	$ \ \{1\}$
\perp	$ \ \{1\}$
$\neg q(y)$	$ \ \{2\}$
\perp	$ \ \{2\}$

SAT

$1 \vee 2$
$\neg 1$
$\neg 2$

Components

$1 \mapsto \neg p(x)$
$2 \mapsto \neg q(y)$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

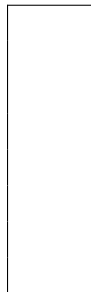
- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO



SAT



Components



Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

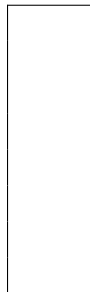
- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

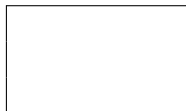
FO



SAT



Components



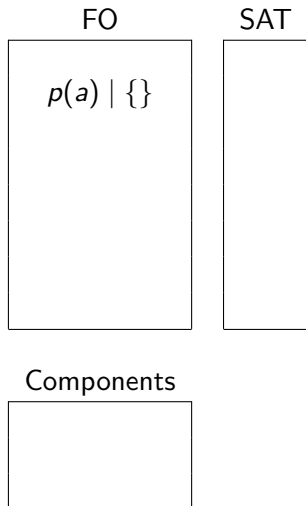
Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- **FO: Process new clauses**
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation



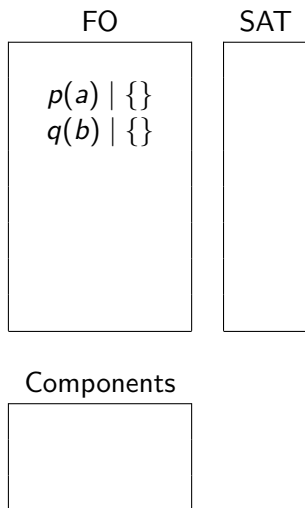
Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- **FO: Process new clauses**
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation



Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$p(a) \mid \{\}$ $q(b) \mid \{\}$

SAT

$1 \vee 2$

Components

$1 \mapsto \neg p(x)$ $2 \mapsto \neg q(y)$
--

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

SAT

$$\vee 2$$

$$3 \vee 2$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model**
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

SAT

$$\underline{1} \vee 2$$

$$\underline{3} \vee 2$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\neg p(x) \vee s(x) \mid \{3\}$$

SAT

$$\underline{1} \vee 2$$

$$\underline{3} \vee 2$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving**
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\neg p(x) \vee s(x) \mid \{3\}(1)$$

SAT

$$\underline{1} \vee 2$$

$$\underline{3} \vee 2$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\neg p(x) \vee s(x) \mid \{3\}(1)$$

$$\perp \mid \{1\}$$

SAT

$$\underline{1} \vee 2$$

$$\underline{3} \vee 2$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\neg p(x) \vee s(x) \mid \{3\}(1)$$

$$\perp \mid \{1\}$$

SAT

$$\underline{1} \vee 2$$

$$\underline{3} \vee 2$$

$$\neg 1$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model**
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\cancel{\neg p(x) \vee s(x)} \mid \{3\}(1)$$

$$\perp \mid \{1\}$$

$$\neg q(y) \mid \{2\}$$

SAT

$$1 \vee \underline{2}$$

$$\underline{3} \vee 2$$

$$\underline{\neg 1}$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \mid \{1\}$$

$$\cancel{\neg p(x) \vee s(x)} \mid \{3\}(1)$$

$$\perp \mid \{1\}$$

$$\neg q(y) \mid \{2\}$$

SAT

$$1 \vee \underline{2}$$

$$\underline{3} \vee 2$$

$$\underline{\neg 1}$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Varying the Example

- Input:

$$p(a), q(b), \neg p(x) \vee \neg q(y), \\ \neg p(x) \vee s(x) \vee \neg q(y)$$

- Repeat

- FO: Process new clauses
 - split clauses into components
- SAT: Construct model
- FO: Use model (do splitting)
- FO: Do FO proving
 - Process refutation

FO

$$p(a) \mid \{\}$$

$$q(b) \mid \{\}$$

$$\neg p(x) \vee s(x) \mid \{3\}$$

$$\neg q(y) \mid \{2\}$$

SAT

$$1 \vee \underline{2}$$

$$\underline{3} \vee 2$$

$$\underline{\neg 1}$$

Components

$$1 \mapsto \neg p(x)$$

$$2 \mapsto \neg q(y)$$

$$3 \mapsto \neg p(x) \vee s(y)$$

Not all vampires are equal

- Vampire uses many strategies in a portfolio mode
- Different strategies are suited to different problems
- In this year's CASC vampire used 152 strategies to solve problems and tried 351
- Further observation: not all strategies needed, but led to quicker proofs
- Intuition: if this applies to problems in general it should apply to these sub-problems

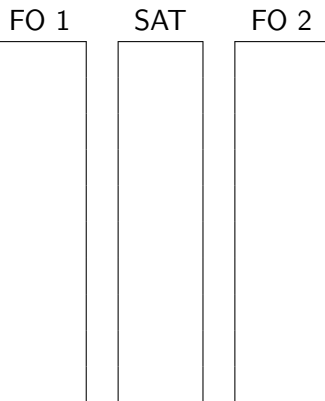
Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO: Process clauses
- 2 SAT: Find model
- 3 FO: Use model
- 4 FO: Do FO proving
 - Process refutation



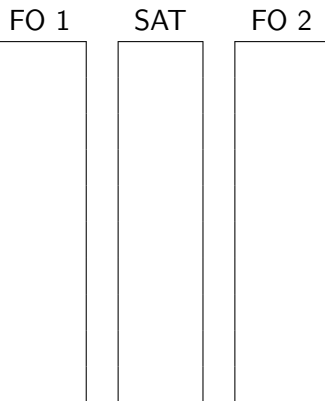
Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO: Process clauses
- 2 SAT: Find model
- 3 FO: Use model
- 4 FO: Do FO proving
 - Process refutation



Example

- Input:

$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$

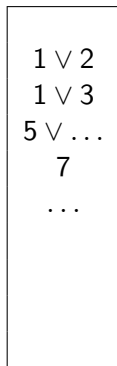
- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

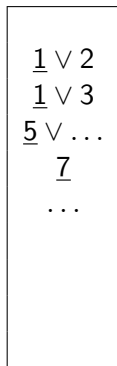
- Repeat

- 1 FO 1: Process clauses
- 2 **SAT: Find model**
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 **FO 1: Use model**
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_1 \mid \{1\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

SAT

$$\underline{1} \vee 2$$

$$\underline{1} \vee 3$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_1 \mid \{1\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

$$\dots$$

$$\perp \mid \{1, 5\}$$

SAT

$$\underline{1} \vee 2$$

$$\underline{1} \vee 3$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

$$\dots$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1	SAT	FO 2
$D_1 \mid \{1\}$	$\underline{1} \vee 2$	
$D_5 \mid \{5\}$	$\underline{1} \vee 3$	
$D_7 \mid \{7\}$	$\underline{5} \vee \dots$	
\dots	$\underline{7}$	
$\perp \mid \{1, 5\}$	\dots	
	$\neg 1 \vee \neg 5$	

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses

- 2 **SAT: Find model**

- 3 FO 1: Use model

- 4 FO 1: Do FO proving

- Process refutation

FO 1

$$D_1 \mid \{1\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$\perp \mid \{1, 5\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

7

...

$$\underline{\neg 1} \vee \neg 5$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 **FO 1: Use model**
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 **FO 1: Do FO proving**
 - Process refutation

FO 1

D_5	$\{5\}$
D_7	$\{7\}$
...	
D_2	$\{2\}$
D_3	$\{3\}$
...	

SAT

$1 \vee \underline{2}$
$1 \vee \underline{3}$
$\underline{5} \vee \dots$
$\underline{7}$
...
$\underline{\neg 1} \vee \neg 5$

FO 2

--

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

...

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

...

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 **FO 1: Process clauses**
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

...

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

...

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 2: Process clauses
- 2 SAT: Find model
- 3 **FO 2: Use model**
- 4 FO 2: Do FO proving
 - Process refutation

FO 1

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

...

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

...

FO 2

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

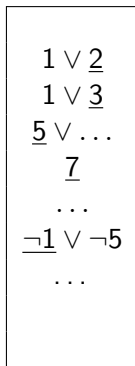
- Repeat

- 1 FO 2: Process clauses
- 2 SAT: Find model
- 3 FO 2: Use model
- 4 FO 2: Do FO proving
 - Process refutation

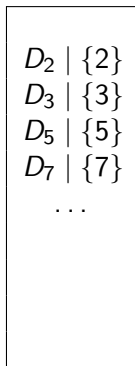
FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

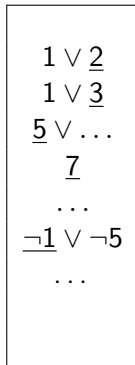
- Repeat

- 1 FO 2: Process clauses
- 2 SAT: Find model
- 3 FO 2: Use model
- 4 FO 2: Do FO proving
 - Process refutation

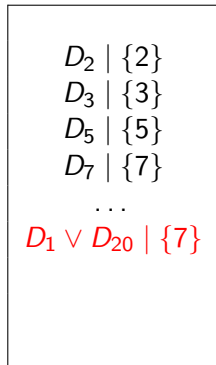
FO 1



SAT



FO 2



Example

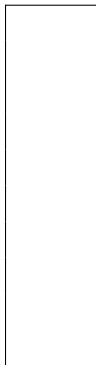
- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

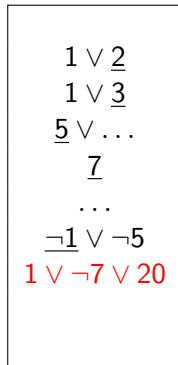
- Repeat

- 1 FO 2: Process clauses
- 2 SAT: Find model
- 3 FO 2: Use model
- 4 FO 2: Do FO proving
 - Process refutation

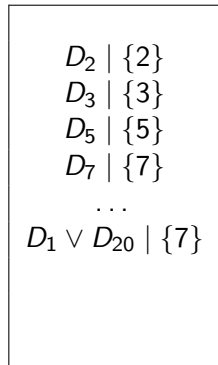
FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 2: Process clauses

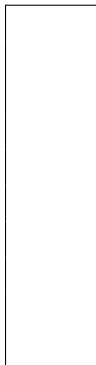
- 2 **SAT: Find model**

- 3 FO 2: Use model

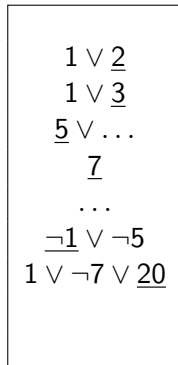
- 4 FO 2: Do FO proving

- Process refutation

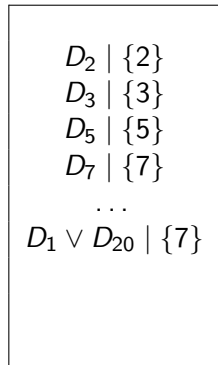
FO 1



SAT



FO 2



Example

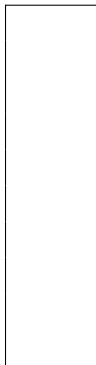
- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

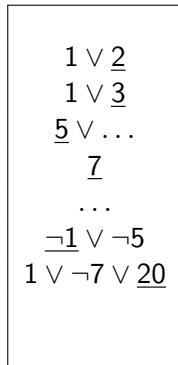
- Repeat

- 1 FO 2: Process clauses
- 2 SAT: Find model
- 3 **FO 2: Use model**
- 4 FO 2: Do FO proving
 - Process refutation

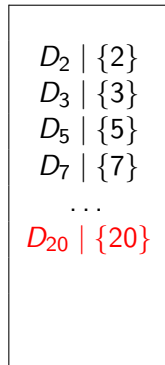
FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 2: Process clauses

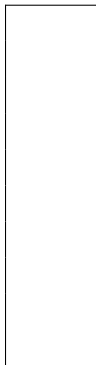
- 2 SAT: Find model

- 3 FO 2: Use model

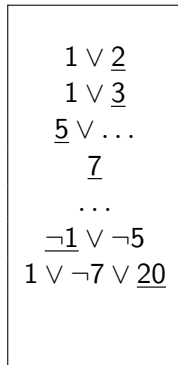
- 4 **FO 2: Do FO proving**

- Process refutation

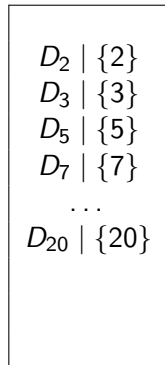
FO 1



SAT



FO 2



Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 **FO 1: Use model**
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_{20} \mid \{20\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

$$1 \vee \neg 7 \vee \underline{20}$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses

- 2 SAT: Find model

- 3 FO 1: Use model

- 4 **FO 1: Do FO proving**

- Process refutation

FO 1

$$D_{20} \mid \{20\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

$$\perp \mid \{7, 20\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

$$1 \vee \neg 7 \vee \underline{20}$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_{20} \mid \{20\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

$$\perp \mid \{7, 20\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

$$1 \vee \neg 7 \vee \underline{20}$$

$$\neg 7 \vee \neg 20$$

FO 2

Example

- Input:

$$D_1 \vee D_2, D_1 \vee D_3, C_1, \dots, C_n$$

- Repeat

- 1 FO 1: Process clauses
- 2 SAT: Find model
- 3 FO 1: Use model
- 4 FO 1: Do FO proving
 - Process refutation

FO 1

$$D_{20} \mid \{20\}$$

$$D_5 \mid \{5\}$$

$$D_7 \mid \{7\}$$

...

$$D_2 \mid \{2\}$$

$$D_3 \mid \{3\}$$

$$\perp \mid \{7, 20\}$$

SAT

$$1 \vee \underline{2}$$

$$1 \vee \underline{3}$$

$$\underline{5} \vee \dots$$

$$\underline{7}$$

...

$$\underline{\neg 1} \vee \neg 5$$

$$1 \vee \neg 7 \vee \underline{20}$$

$$\neg 7 \vee \neg 20$$

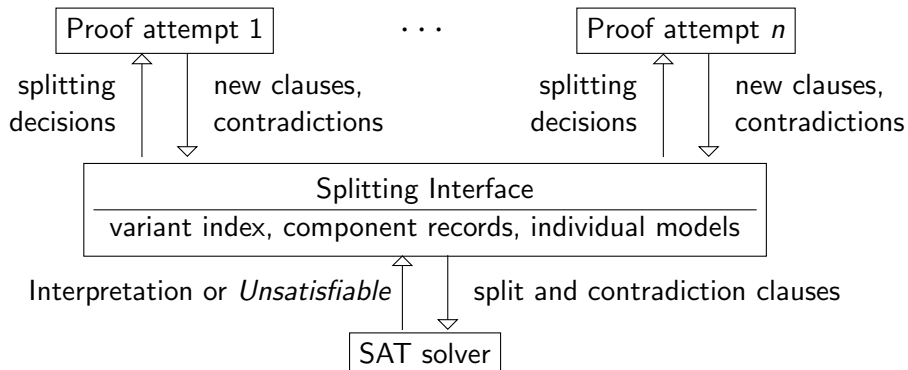
FO 2

- SAT Refutation

Organising many proofs

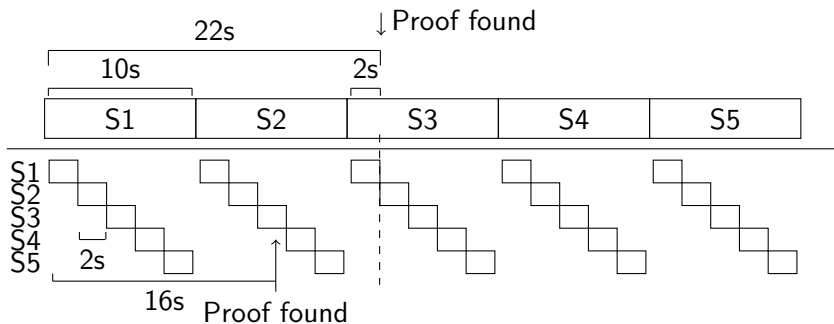
- Some technical stuff to solve
- Firstly, this required lots of reorganisation of the vampire architecture to separate data structures etc.
- How to deal with strategies that alter problem?
 - Currently restrict cooperating strategies to same preprocessed problem
- How to switch between proof attempts?
 - We introduce an interleaving architecture

Shared AVATAR Architecture



Interleaving Strategies

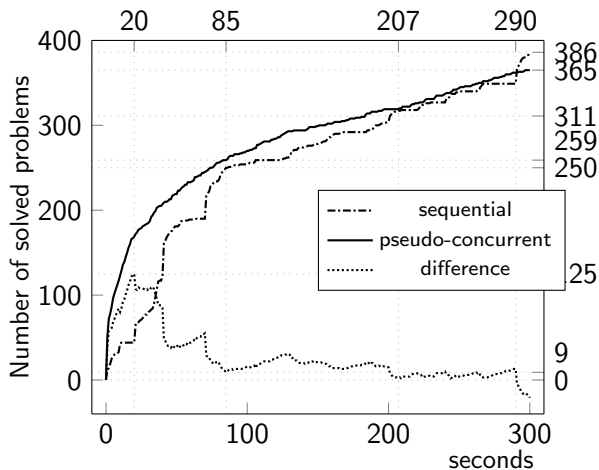
- Generally if a strategy finds a proof it finds it quickly
- By interleaving strategies we can find the quick proofs faster



Some results...

- ... but the whole architecture is currently being rewritten to remove inefficiencies
- We took
 - 1747 very hard first-order problems from TPTP
 - 30 random 'sensible' strategies
- And ran
 - Each strategy independently for 10 seconds
 - All 30 together with a per-strategy 10 second time limit
- We found
 - Problems were solved on average 1.53 times faster
 - Sharing splitting decisions led to 63 more problems being solved, often quickly.
 - It also solved some rating 1 problems
 - However some problems were lost. There are two explanations
 - SAT solver overhead goes up 20%
 - Loss of memory locality

Experiment



Conclusion

- AVATAR uses SAT models to guide proof search
- The changing models represent different sub-problems
- As different strategies are good for different problems it makes sense to solve these sub-problems as separate proof attempts
- Some more engineering required before doing more exciting things
- Another exciting extension...
 - Replace SAT solver by SMT solver
 - We've done this for the single proof attempt version with Z3

VampireZ3... extra bits

- If a component is ground then do not name it, instead translate it to Z3 syntax
- If it uses interpreted operations or numbers, translate these into the appropriate Z3 operations or numbers
- That's it
- Z3 will only produce models consistent with the underlying theories
- The FO solver only needs to consider this (much) smaller set of sub-problems
- Note: Only ground bits go to Z3, we still need to do non-ground theory reasoning