

Natural deduction



$$\begin{array}{c}
 \frac{}{\Gamma, A \vdash A} \text{ Assume} \\
 \\
 \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge I \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge E_1 \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge E_2 \\
 \\
 \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee I_1 \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee I_2 \quad \frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash A \vee B \vdash C} \vee E \\
 \\
 \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow I \quad \frac{\Gamma \vdash A \quad \Gamma \vdash A \Rightarrow B}{\Gamma \vdash B} \Rightarrow E
 \end{array}$$

Natural deduction



$$\begin{array}{c}
 \frac{\Gamma, A \vdash F}{\Gamma \vdash \neg A} \neg I \quad \frac{\Gamma \vdash \neg \neg A}{\Gamma \vdash A} \neg E \\
 \\
 \frac{\Gamma \vdash A \quad \Gamma \vdash \neg A}{\Gamma \vdash F} FI \quad \frac{\Gamma \vdash F}{\Gamma \vdash A} FE \\
 \\
 \frac{}{\Gamma \vdash T} TI \quad \text{No T elimination}
 \end{array}$$

Note: one can get rid of the FE without losing expressiveness. Can someone see why?

Two kinds of non-determinism



- Don't care non-determinism (also called conjunctive non-determinism)
 - All choices will lead to a successful search, so we "don't care" which one we take
- Don't know non-determinism (also called disjunctive non-determinism)
 - Some of the choices will lead to a successful search, but we "don't know" which one a priori
- We will see several techniques for reducing non-determinism

1. Validity preservation



- An inference rule is said to preserve validity if its premise judgments are valid iff its conclusion judgment is
- One direction of this definition (namely, "premises" valid implies "conclusion" valid) always holds because it is required by soundness.
- The other direction may not hold

1. Validity preservation



- When doing a backward search, rules that preserve validity cannot be the source of "don't know" non-determinism
- Among validity preservation rules, it doesn't matter which one is applied or how it is instantiated, because none of them will lead the search astray.
- Another way of looking at it is that if all the rules are validity preserving, then there is no need to backtrack
 - Notice the parallel with a forward search, where we don't need to backtrack because all rules are sound

1. Validity preservation



- Great, so which rules in natural deduction preserves validity?
- Not many...
- The sequent calculus will improve on this as most of its rules (and in some versions, all of its rules) will be validity preserving

2. Sub-formula property

$\left[\begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \right]$
 $\Rightarrow \Gamma \vdash A \Leftarrow$

- A proof system has the sub-formula property if the derivation of a judgment J only contains sub-formulas of the formulas occurring in J
 - More precisely, a proof system has the sub-formula property if, whenever there exists a derivation of a judgment J, then there exists a derivation of J that only contains sub-formulas of the formulas in J
- This property is useful because it suggests ways of instantiating free meta-variables in a forward or a backward search: use sub-formulas of the goal J.

2. Sub-formula property

- Prove $A \vee \neg A$ in natural deduction

$\frac{\Gamma \vdash A \quad \text{Assume}}{\Gamma \vdash A \vee \neg A} \vee I$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \quad \text{Assume}}{\Gamma \vdash \neg(A \vee \neg A)} \neg I$
 $\frac{\Gamma \vdash \neg(A \vee \neg A), \neg(A \vee \neg A) \vdash F}{\Gamma \vdash \neg(A \vee \neg A) \vdash \neg(A \vee \neg A)} \neg E$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \vdash \neg(A \vee \neg A)}{\Gamma \vdash \neg(A \vee \neg A) \vdash F} \neg E$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \vdash F}{\Gamma \vdash A \vee \neg A} \vee I$

Cheat sheet

$\frac{}{\Gamma, A \vdash A} \text{Assume}$
 $\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge I$
 $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge E_1$
 $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge E_2$
 $\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee I_1$
 $\frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee I_2$
 $\frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C \quad \Gamma \vdash A \vee B}{\Gamma \vdash C} \vee E$
 $\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow I$
 $\frac{\Gamma \vdash A \quad \Gamma \vdash A \Rightarrow B}{\Gamma \vdash B} \Rightarrow E$

Cheat sheet

$\frac{\Gamma, A \vdash F}{\Gamma \vdash \neg A} \neg I$
 $\frac{\Gamma \vdash \neg \neg A}{\Gamma \vdash A} \neg E$
 $\frac{\Gamma \vdash A \quad \Gamma \vdash \neg A}{\Gamma \vdash F} F I$
 $\frac{\Gamma \vdash F}{\Gamma \vdash A} F E$
 $\frac{}{\Gamma \vdash T} T I$
 No T elimination

2. Sub-formula property

- Prove $A \vee \neg A$ in natural deduction

$\frac{\Gamma \vdash A \quad \text{Assume}}{\Gamma \vdash A \vee \neg A} \vee I$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \quad \text{Assume}}{\Gamma \vdash \neg(A \vee \neg A)} \neg I$
 $\frac{\Gamma \vdash \neg(A \vee \neg A), \neg(A \vee \neg A) \vdash F}{\Gamma \vdash \neg(A \vee \neg A) \vdash \neg(A \vee \neg A)} \neg E$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \vdash \neg(A \vee \neg A)}{\Gamma \vdash \neg(A \vee \neg A) \vdash F} \neg E$
 $\frac{\Gamma \vdash \neg(A \vee \neg A) \vdash F}{\Gamma \vdash A \vee \neg A} \vee I$

2. Sub-formula property

- Natural deduction, in its vanilla flavor, does not have the sub-formula property
- Proofs take "detours" through formulas that are not in the goal
- The (cut-free) sequent calculus will improve on this, as it will have the sub-formula property

3. Picking the right direction



- Which direction should we apply these rules in?

$$\begin{array}{c}
 \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge I \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge E_1 \quad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge E_2 \\
 \uparrow \frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee I_1 \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee I_2 \quad \downarrow \frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash A \vee B} \vee E \\
 \frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow I \quad \frac{\Gamma \vdash A \quad \Gamma \vdash A \Rightarrow B}{\Gamma \vdash B} \Rightarrow E
 \end{array}$$

3. Picking the right direction



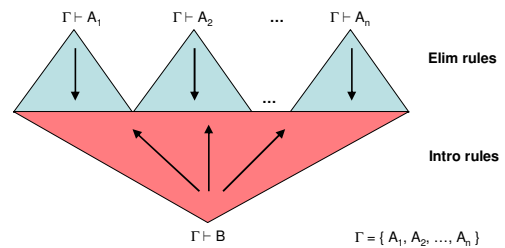
- The amount of non-determinism in instantiating a rule depends on the direction in which the rule is applied
 - If the premise of a rule has more meta-variables than the conclusion, then forward application of the rule will require less choice than the backward application
 - Symmetrically, if the conclusion has more meta-variables than the premise, then the forward application will have more non-determinism
- The sequent calculus will improve on natural deduction in this respect, since most of its rules will have the same meta-variables in the premise and the conclusion (the only exception being quantifier rules)

3. Picking the right direction



- Introduction rules are more deterministic going backward
- Elimination rules are more deterministic going forward
- This suggests an approach based on bi-directional search: introduction rules are used in the backward direction, elimination rules are used in the forward direction, and a derivation is found when the two meet
- The final derivation uses elimination rules to split assumptions up into their constituent components and then introduction rules to combine these constituent components into the goal formula

3. Picking the right direction



3. Picking the right direction



- For the intuitionistic subset of natural deduction, all proofs can be expressed in such a form
 - This does not hold for the classical version of natural deduction
- The intercalation calculus of John Byrnes embodies this idea for intuitionistic natural deduction
- The sequent calculus will be of help for this approach, since it will simulate the bi-directional search of intercalation using only a backward search

Summary



- Three techniques to reduce non-determinism:
 - Validity preservation
 - Sub-formula property
 - Pick the right direction to apply rules

Sequent calculus



- Sequent calculus developed in 1935 by Gentzen in the same seminal paper as natural deduction
 - Coincidentally, this paper also introduces the \forall notation for universal quantifiers
- Sequents were originally introduced as a device for proving natural deduction consistent
 - Natural deduction corresponds to the way humans reason, but proofs in natural deduction are sometimes long and indirect
 - Proofs in the sequent calculus are much more direct, and this directness property allowed Gentzen to show consistency of sequents
 - Natural deduction was then shown consistent by demonstrating its equivalence to the sequent calculus

Sequent calculus



- Although the original motivation for sequents had nothing to do with automated reasoning, the more direct form of proofs makes it a good fit for automated theorem provers
- Sequents are also fairly intuitive to understand, and so they provide a good balance between human friendliness and machine friendliness
- Sequent style systems are used in PVS and Twelf

Sequent calculus



- A judgment in the sequent calculus has the form $\Gamma \vdash \Delta$
- The interpretation is that the conjunction of formulas in Γ entails the disjunction of formulas in Δ
- In other words: $\{\alpha_1, \dots, \alpha_n\} \vdash \{\beta_1, \dots, \beta_m\}$ is interpreted as $\alpha_1 \wedge \dots \wedge \alpha_n \Rightarrow \beta_1 \vee \dots \vee \beta_m$

Sequent calculus



$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B}$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B}$$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B}$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$$

Sequent calculus



$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B} \vee_1 \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B} \vee_2 \quad \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \vee R$$

$$\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge I \quad \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B} \wedge R$$

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \Rightarrow B} \Rightarrow I \quad \frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B} \Rightarrow R$$

$$\frac{\Gamma, A \vdash F}{\Gamma \vdash \neg A} \neg I \quad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \neg R$$

Right rules correspond to introduction rules in natural deduction

Sequent calculus



$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee L$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \neg L$$

Sequent calculus



$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee L \qquad \frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma \vdash C} \vee E$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge E_1 \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge E_2$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L \qquad \frac{\Gamma \vdash A \quad \Gamma \vdash A \Rightarrow B}{\Gamma \vdash B} \Rightarrow E$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \neg L \qquad \frac{\Gamma \vdash \neg \neg A}{\Gamma \vdash A} \neg E$$

Left rules correspond to the inverse application of ND elim rules

Sequent calculus: logical rules



$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee L \qquad \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \vee R$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L \qquad \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B} \wedge R$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L \qquad \frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B} \Rightarrow R$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \neg L \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \neg R$$

Sequent calculus: axioms



$$\frac{}{A \vdash A} \text{ Assume}$$

- Note how this is different from the natural deduction axiom:

$$\frac{}{\Gamma, A \vdash A} \text{ Assume}$$

Sequent calculus: structural rules



Thinning

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}$$

Contraction

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

$$\frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}$$

Exchange

$$\frac{\Gamma, A, B, \Pi \vdash \Delta}{\Gamma, B, A, \Pi \vdash \Delta}$$

$$\frac{\Gamma \vdash \Delta, A, B, \Pi}{\Gamma \vdash \Delta, B, A, \Pi}$$

Cut rule



$$\frac{\Gamma \vdash \Delta, A \quad A, \Pi \vdash \Lambda}{\Gamma, \Pi \vdash \Delta, \Lambda}$$

$A \Rightarrow B \quad B \Rightarrow C$
 $A \Rightarrow C$

- Gentzen's cut-elimination theorem (also called Hauptsatz): if there exists a derivation of a judgment J in the sequent calculus that uses the cut rule, then there also exists a derivation of J that does not use the cut rule

Cut rule: why have it there in the first place?

- It can make proofs smaller
 - There are examples where the proof using cut takes a page, but the cut-free proof cannot be computed in the lifespan of the universe!
- It makes proving equivalence with respect to natural deduction easier

Cut rule: why remove it?



- After removing the cut rule, all rules in the sequent calculus (except for quantifier rules) have the same number of meta-variables in the premise and the conclusion
- In terms of automation, this property is an improvement over natural deduction, where introduction rules were easier to apply in the backward direction, and elimination rules were easier to apply in the forward direction

Cut rule: why remove it?



- Removing the cut rule makes the sequent calculus have the sub-formula property
 - don't need to take detours through formulas that are not mentioned in the original goal
 - In general, this makes proofs smaller and more direct (although in the worst case, it can make proofs be gigantic)
- Revisiting our previous example:
 - prove $A \vee \neg A$

Cheat sheet



$$\frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee L$$

$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \vee R$$

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \wedge B} \wedge R$$

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L$$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B} \Rightarrow R$$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \neg L$$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \neg R$$

Other properties of sequents



- In the cut-free version of the sequent calculus presented here, all rules are validity preserving
 - This removes the need to backtrack
- Also, because applying left rules corresponds to the inverse application of elim rules, Byrnes bi-directional approach can be seen as a unidirectional search in the sequent calculus

Summary of sequents



- Same number of meta-variables in premise and consequents (except for quantifier rules)
 - Reduces non-determinism
- Sub-formula property
 - Makes proofs more direct
- Validity preservation (in some versions)
 - Don't need to backtrack
- Can do natural-deduction-bi-directional searches in one direction

Tactics and Tacticals



- LCF: Logic for Computable functions, 1969
- A “meta-language” for manipulating proofs
 - The language of proofs: higher-order logic
 - The meta-language for manipulating proofs: functional language with static types to help make sure that the proofs are manipulated correctly
- This language was called ML for meta-language
- LCF introduced tactics and tacticals, but variants have appeared in many systems, including HOL, PVS, Isabelle and NuPRL

Tactics



- A tactic is a backward application of an inference rule
 - More broadly, a tactic is a way of decomposing a goal into sub-goals while performing a backward search
- When applied, a tactic can either succeed
 - in which case it create a sub-goal
- Or it can fail because it is not applicable
 - in this case, the goal is not changed

Tactics



- The system provides a set of primitive tactics, for example:
 - skip: always success, does nothing
 - fail: always fails
 - apply- \Rightarrow L: apply the \Rightarrow L rule
 - skolemize: if the formula to prove is a universal, remove the quantifier
 - simplify: simplify based on decision procedures
 - Induct: if the formula to prove is a universal, try to apply induction
 - ... (a typical system provides many more)

Tacticals



- A tactical is a function that takes tactics and returns tactics
- The system includes a set of predefined tacticals, for example:
 - IF-THEN-ELSE: takes three tactics a, b, and c, and returns a tactic that applies a to the goal, and, if a succeeds, it then applies b to all sub-goals created by a, otherwise, it applies c to all sub-goals created by a
 - SEQ: takes two tactics a and b, and returns a tactic that applies a to the goal, and b to all the sub-goal produced by a
 - REPEAT: takes a given tactic, and repeatedly applies it until it fails

Tacticals



- Using these predefined tacticals, programmers can create new tactics from primitive tactics
- They can also create their own tacticals
- Tacticals can be seen as creating more complicated inference steps from simpler ones

Examples



```
;; A tactic that repeatedly skolemizes
;; and then simplifies
define grind =
  REPEAT(SEQ(skolemize, simplify))
```

Examples



```
;; A tactical that tries all the tactics
;; in the given list in sequence. If
;; one of the tactic succeeds, that
;; tactic is applied and the try succeeds.
;; Otherwise, the try fails
define try(list) =
  case list of
  nil: fail
  first::rest:
    IF-THEN-ELSE(first,
                  skip,
                  try(rest))

;; Perform a backward search, by first applying
;; introduction rules repeatedly, and then applying
;; elimination rules to the results
define backsearch = SEQ(
  REPEAT(try(list-of-I-rules)),
  REPEAT(try(list-of-E-rules)))
```

Note: This is not the syntax of any particular system. Each system has its own syntax and language for expressing tactics and tacticals. For example, LCF and its successors (such as HOL) use ML. PVS uses LISP.

Tactics and tacticals: Summary



- Tactics and tacticals provide a way for a programmer to customize a theorem prover
- Many variants exist, and each system has its own primitives and its own language for defining tactics and tacticals