Coherent Logic — an overview

Marc Bezem
Department of Informatics
University of Bergen
(github.com/marcbezem/CL-PC22)

September 2022

Crash course in Coherent Logic (CL)

- Basics
- Proof theory for CL
- Metatheory
- Translation from FOL to CL
- Evaluation of CL as a fragment of FOL

Automated reasoning in CL

- Automated reasoning
- Elimination of function symbols
- Proof search strategies

Applications of CL

- Proof assistants
- Model finding
- Constructive algebra

Coherent logic preliminaries 1

- Fix a finite first-order signature Σ
- Positive formulas: built up from atoms using $\top, \bot, \vee, \wedge, \exists$
- Coherent implications (sentences): $\forall \vec{x}.\ (C \to D)$ with C, D positive formulas
- Coherent theory: axiomatized by coherent sentences
- $\vee\exists\wedge$-formula: $(\exists \vec{y}_1 . A_1) \vee \cdots \vee (\exists \vec{y}_k . A_k)$, $k \geq 0$, with each A_i a (possibly empty) conjunction of atoms
- Lemma 1. Every positive formula is (constructively) equivalent to a $\vee\exists\wedge$-formula. Proof by induction:
  - Base cases: atom (one disjunct, empty $\exists$, one conjunct); $\bot$ (empty $\vee$); $\top$ (one disjunct with empty $\exists, \wedge$)
  - Induction cases: $\vee$ (trivial); $\wedge$ (distributivity + $(\exists x.\varphi) \wedge (\exists y.\psi)$ iff $\exists xy.\ (\varphi \wedge \psi)$); $\exists$ (commutes with $\vee$)

Coherent logic preliminaries 2

- Lemma 2. Every coherent implication is (constructively) equivalent to a finite set of coherent implications $\forall \vec{x}.\ (C \to D)$ with C a conjunction of atoms and D a $\vee\exists\wedge$-formula
- Proof. Use Lemma 1 to replace C and D by $\vee\exists\wedge$-formulas. Then use $(\varphi \vee \psi) \to D$ iff $(\varphi \to D) \wedge (\psi \to D)$, and $(\exists y.\varphi) \to D$ iff $\forall y.\ (\varphi \to D)$
- Notation: we use the format of Lemma 2, leaving out the universal prefix, and omitting the premiss '$C \to$ ' if $C \equiv \top$
- Discuss: $\exists y.\top$ and $\exists y.\bot$ and $\forall y.\top$ and $\forall y.\bot$
- Full compliance with Tarski semantics if Σ has a constant

Examples

- all usual equality axioms, including congruence
- $p \vee np$ and $p \wedge np \to \bot$ (NB $p \vee \neg p$ is not coherent)
- lattice theory: $\exists z.\ meet(x, y, z)$
- geometry: $p(x) \wedge p(y) \to \exists z.\ \ell(z) \wedge i(x, z) \wedge i(y, z)$
- rewriting, $\diamond$-property: $r(x, y) \wedge r(x, z) \to \exists u.\ r(y, u) \wedge r(z, u)$
- weak-tc-elim: $r^*(x, y) \to (x = y) \vee \exists z.\ r(x, z) \wedge r^*(z, y)$
- seriality: $\exists y.\ s(x, y)$ (who needs a function?)
- field theory: $(x = 0) \vee \exists y.\ (x \cdot y = 1)$
- local ring: $\exists y.\ (x \cdot y = 1) \vee (\exists y.\ ((1 - x) \cdot y = 1)$ (equivalent to the more common: if $x + y$ is a unit, then x is a unit or y is a unit).

History of CL

- Skolem (1920s): coherent formulations of lattice theory and projective geometry, calling the axioms “Erzeugungsprinzipien" (production rules), anticipating ground forward reasoning. Using CL,
  - Skolem solved a decision problem in lattice theory
  - Skolem gave a method to test in/dependence from the axioms of plane projective geometry (example: Desargues’ Axiom)
- Grothendieck (1960s): geometric morphisms preserve geometric logic (= coherent logic + infinitary disjunction). This is quite complicated, but we’ll see a glimpse in the forcing semantics of coherent logic given later.

A proof theory for CL

- In short: ground forward reasoning with case distinction and introduction of witnesses (ground tableau reasoning)
- In full: define inductively $\Gamma \vdash^T_{\vec{y}} A$, where A (Γ) atom (set of atoms) with all variables in $\vec{y}$, in case

(base) A is in Γ, or in case

(step) T has an axiom $\forall \vec{x}.\ (C \to (\exists \vec{y}_1 . B_1) \vee \cdots \vee (\exists \vec{y}_k . B_k))$ such that for some sequence of terms $\vec{t}$ with variables in $\vec{y}$ we have

  - $C[\vec{t}/\vec{x}]$ is a subset of Γ, and
  - $\Gamma, B_i[\vec{t}/\vec{x}] \vdash^T_{\vec{y},\vec{y_i}} A$ for all $i = 1, \ldots, n$ (NB $\vec{y_i}$ fresh wrt $\vec{y}$)

- Rough visualization as a tree with inner nodes like

$$\frac{\Gamma, B_1[\vec{t}/\vec{x}] \quad \cdots \quad \Gamma, B_n[\vec{t}/\vec{x}]}{\Gamma}\ axiom$$

- NB we omit conclusion A in all the nodes, but we should actually keep track of the $\vec{y}, \vec{y_i}$. Looking ahead, pairs like $(\vec{y}; \Gamma)$ will be the forcing conditions, $\approx$ finite Kripke worlds.

Derivation trees in CL, example and general procedure

- Let T consists of $p \vee \exists x.\ q(x)$ and $p \to \bot$ and $q(y) \to r$
- Derivation tree for $\emptyset \vdash^T_\emptyset r$

$$\dfrac{\dfrac{(\bot)}{\{p\}}\ p \to \bot \qquad \dfrac{\{q(c), r\}}{\{q(c)\}}\ q(y) \to r}{\emptyset}\ p \vee \exists x.\ q(x)$$

- Tree construction: from $\emptyset$, repeat exhaustively 1,2,3 below
  1. Pick a leaf node ($\neq (\bot)$) without A in its Γ (else done)
  2. Pick fairly a Γ-false instance of an axiom of T (else fail: Γ is a model of T not containing A, so A is underivable)
  3. Extend the tree in the leaf node according to the instance
- Fairness is tricky to define, but crucial for the following completeness result (to be proved on the next slide):
- The tree construction stops in 1 iff A is derivable (if-part!)
- Example for explaining un/fairness: $\exists y.s(x, y)$ and $p(0)$

Soundness and completeness wrt Tarski semantics

- Soundness easily proved by induction on $\Gamma \vdash^{T}_{\vec{y}} A$
- Not complete: $\emptyset \vdash^{\forall x.\bot}_{\emptyset} p$ underivable without a constant in Σ
- Silly, let's assume a constant in Σ, or just $\exists x.\top$
- Proof of completeness: essentially non-constructive. Assume $\forall \vec{y}.\ (\Gamma \to A)$ holds in any model of T. Build the tree for $\Gamma \vdash^{T}_{\vec{y}} A$. Recall the that the sets Γ grow along the branches. If the tree is finite, it is a proof (2 cannot happen). If not, it has an infinite branch by König's Lemma. Collect the set of variables Y and the set of atoms M along the infinite branch. Build a model with domain $\mathrm{Tm}^{\Sigma}(Y)$ and positive diagram M. This is a model of T (by fairness) containing Γ but not A. Contradiction.
- Proof theory easily extended to arbitrary coherent conclusions of a coherent theory T.

Metatheoretic results and remarks

- Corollary of completeness: given a coherent theory T, classically provable coherent sentences are constructively provable
- For geometric logic this is called Barr's Theorem (anticipated by Lawvere and Deligne)
- Completeness and Barr's Theorem are not constructive
- Barr's Theorem for coherent logic can be proved constructively using a cut-elimination argument
- Coherent completeness wrt forcing semantics is constructively provable, but does not give the conservativity of classical reasoning
- NB: the forcing semantics is sound wrt cosntructive logic for arbitrary formulas

Translation from FOL to CL

- Idea: introduce two new predicate symbols $T(\psi), F(\psi)$ for each subformula ψ of a given formula φ, with the arities of $T(\psi), F(\psi)$ being the number of free variables of ψ. The rules for signed tableaux are coherent axioms:
  - if $\psi(\vec{x}) \equiv \psi_1 \wedge \psi_2$, then $\begin{cases} T(\psi)(\vec{x}) \to T(\psi_1)(\vec{x}) \wedge T(\psi_2)(\vec{x}) \\ F(\psi)(\vec{x}) \to F(\psi_1)(\vec{x}) \vee F(\psi_2)(\vec{x}) \end{cases}$
  - if $\psi(\vec{x}) \equiv \psi_1 \vee \psi_2$, then ...
  - if $\psi(\vec{x}) \equiv \psi_1 \to \psi_2$, then $\begin{cases} T(\psi)(\vec{x}) \to F(\psi_1)(\vec{x}) \vee T(\psi_2)(\vec{x}) \\ F(\psi)(\vec{x}) \to T(\psi_1)(\vec{x}) \wedge F(\psi_2)(\vec{x}) \end{cases}$
  - if $\psi(\vec{x}) \equiv \neg\psi_1$, then ...
  - if $\psi(\vec{x}) \equiv \forall y.\psi_1(\vec{x}, y)$, then $\begin{cases} T(\psi)(\vec{x}) \to T(\psi_1)(\vec{x}, y) \\ F(\psi)(\vec{x}) \to \exists y.F(\psi_1)(\vec{x}, y) \end{cases}$
  - if $\psi(\vec{x}) \equiv \exists y.\psi_1(\vec{x}, y)$, then $\begin{cases} T(\psi)(\vec{x}) \to \exists y.T(\psi_1)(\vec{x}, y) \\ F(\psi)(\vec{x}) \to F(\psi_1)(\vec{x}, y) \end{cases}$
  - if $\psi(\vec{x})$ is atomic, then $(T(\psi)(\vec{x}) \wedge F(\psi)(\vec{x})) \to \bot$
- By the completeness of signed tableaux: φ is a tautology iff $F(\varphi) \vdash_{\emptyset}^{Coh(\varphi)} \bot$, with $Coh(\varphi)$ all the above axioms

Example in propositional logic: Peirce's Law

- Peirce's Law: $\varphi :\equiv ((p \to q) \to p) \to p$
- To prove: $F(((p \to q) \to p) \to p) \vdash_{\emptyset}^{Coh(\varphi)} \bot$
- Part of $Coh(\varphi)$ that is actually used:
  1. $F(((p \to q) \to p) \to p) \to (T((p \to q) \to p) \wedge F(p))$
  2. $T((p \to q) \to p) \to (F(p \to q) \vee T(p))$
  3. $F(p \to q) \to (T(p) \wedge F(q))$
  4. $(T(p) \wedge F(p)) \to \bot$
- Proof: use 1, 2, 3 and split on $F(p \to q) \vee T(p)$, ...
- Details on the blackboard
- Proof of φ: take $T :\equiv \lambda\varphi.\, \varphi$, $F :\equiv \lambda\varphi.\, \neg\varphi$. Then 1,2,3,4 are easy (but classical), the CL proof is also a proof in propositional logic, and we finish by RAA

Example in predicate logic: the Drinker's Paradox

- Drinker's Paradox: $\varphi :\equiv \exists x.\ (d(x) \to \forall y.d(y))$
- To prove: $F(\exists x.\ d(x) \to \forall y.d(y))) \vdash_{\emptyset}^{Coh(\varphi)} \bot$
- Part of $Coh(\varphi)$ that is actually used:
  1. no take-off without $\exists x.\top$, alternative: prove $F(\varphi) \vdash_{\{c\}}^{Coh(\varphi)} \bot$
  2. $\forall x.\ (F(\exists x.\ d(x) \to \forall y.d(y))) \to F(d(x) \to \forall y.d(y)))$
  3. $\forall x.\ (F(d(x) \to \forall y.d(y)) \to (T(d(x)) \wedge F(\forall y.d(y))))$
  4. $F(\forall y.d(y)) \to \exists y.F(d(y))$
  5. $\forall x.\ (T(d(x)) \wedge F(d(x))) \to \bot$
- Proof: use 1 and get c, instantiate 2 and 3 with c and get $T(d(c)) \wedge F(\forall y.d(y))$, so by 4 we get c' with $F(d(c'))$, ...
- Details on the blackboard
- Proof of φ in FOL: take $T :\equiv \lambda\varphi.\ \varphi$, $F :\equiv \lambda\varphi.\ \neg\varphi$. Then 1–5 are easy (Tarski and classical), the CL proof is also a proof in FOL, and we finish by RAA

Translation from FOL to CL (ctnd)

- Skolem (1920): Every FOL theory has a definitional extension that is equivalent to a $\forall\exists$ theory
- Many variations possible (Polonsky, Dyckhoff & Negri, Fisher, Mints)
- Possible objectives: fewer new predicates, fewer CL axioms ..., keeping a coherent axiom coherent
- Polonsky proposed several improvements, starting from NNF, flipping polarities, also using reversed tableaux rules
- Dyckhoff & Negri: add $T(\psi)(\vec{x}) \to \psi(\vec{x})$ and $(F(\psi)(\vec{x}) \wedge \psi(\vec{x})) \to \bot$ for all atomic ψ and obtain: Every FOL theory has a positive semi-definitional extension that is equivalent to a CL theory
- Consequences in CL are always constructive
- Translation of FOL to CL contains many non-constructive steps, often more than necessary

Evaluation of CL as a fragment of FOL

- Constructive, with classical logic a conservative extension
- Simpler metatheory: proof theory, completeness, conservativity of skolemization (elimination of $\exists$)
- Applications to metamathematics: independence, decision problems
- Other applications:
  - automated reasoning, supporting proof assistants
  - model finding
  - constructive algebra

Automated reasoning (AR)

- We focus on AR in (fragments of) FOL
- There are dozens of FOL provers (Vampire wins CASC)
- TPTP is a large database of AR problems (CNF/FOL/HOL)
- Different branch of AR: model finding (SAT/CNF/FNT)
- There are a handful of CL provers (competitive on CL problems, but not on FOL problems):
  - SATCHMO+ (Bry et al.)
  - Argo, Larus (Janicic et al.)
  - Geo (Nivelle et al.)
  - Colog (Fisher)
  - EYE (De Roo, semantic web)
- Most CL provers support only 0-ary function symbols
- We describe later how to eliminate function symbols

Rationale for automated reasoning in CL

- Expressivity of CL is between CNF (resolution) and FOL
- Different balance: expressivity versus efficiency
- Skolemization (elimination of $\exists$) not necessary
  - Skolemization makes the Herbrand Universe infinite
  - Why skolemize an axiom like $p(x, y) \rightarrow \exists z.\ p(x, z)$?
  - Skolemization changes meaning (problematic for interactive theorem proving, and for obtaining proof objects)
  - Skolem functions obfuscate symmetries (cf. $\diamond$-property)
  - But: skolemized proofs can be exponentially shorter!
- Ground forward reasoning is very simple and intuitive, proof objects can easily be obtained
- But: non-ground proofs can be exponentially shorter!

Elimination of function symbols

- Idea: use the graph instead of the function, i.e., new $(n+1)$-predicates for n-ary functions, for example:
  - For constants: $c(x)$ (expressing $c = x$), axiom $\exists x.\, c(x)$
  - For unary functions: $s(x, y)$ (expressing $s(x) = y$), axiom $\exists y.\, s(x, y)$
- Example: the term $f(s(x), o)$ leads to a condition $s(x, y) \wedge o(u) \wedge f(y, u, z)$ after which every occurrence of $f(s(x), o)$ is replaced by z. Then $\forall \vec{x}.\ (C \to D)$ becomes $\forall x, y, u, z, \vec{x}\ (s(x, y) \wedge o(u) \wedge f(y, u, z) \wedge C' \to D')$ where C', D' are the result of the substitution in C and D.
- Example: $a = b$ becomes $a(x) \wedge b(y) \to x = y$
- Unicity, e.g., $c(x) \wedge c(y) \to x = y$, not required! (since the new conditions only occur in negative positions)

Puzzle, formalized in CL with functions (Nivelle)

- Can one color each $n \in \mathbb{N}$ either red or blue but not both such that, if n is red, then $n+2$ is blue, and if n is blue, then $n+1$ is red?
- No: consider $0?23\ldots$ and $01?34\ldots$
- CL theory:
  1. $r(x) \vee g(x)$
  2. $r(x) \wedge g(x) \rightarrow \bot$
  3. $r(x) \rightarrow g(s(s(x)))$
  4. $g(x) \rightarrow r(s(x))$
- Do we miss something?
- Yes, domain non-empty:
  5. $\exists x.\ \top$

Puzzle, function eliminated

- See `LABresources/hdn.in`
  1. $r(x) \vee g(x)$
  2. $r(x) \wedge g(x) \rightarrow \bot$
  3. $r(x) \wedge s(x,y) \wedge s(y,z) \rightarrow g(z)$
  4. $g(x) \wedge s(x,y) \rightarrow r(y)$
  5. $\exists x.\ \top$
  6. $\exists y.\ s(x,y)$
- Solution of version of puzzle with the function:
  - Note that the substitution principle is valid
  - Substitute $(s(x) = y)$ for $s(x,y)$ in 3,4,6:
    - Regarding 6, $\exists y.\ s(x) = y$ is trivial
    - Regarding 4, $g(x) \wedge s(x) = y \rightarrow r(y)$ is equivalent to $g(x) \rightarrow r(s(x))$
    - Similarly for 3 (and, in general, for any function)

Depth-first proof search in CL

- Recall the tree construction on slide 8
- Any open leaf is fine, so we always take the leftmost
- What instance of which Γ-false axiom to pick?
- NB two trees: derivation tree and the search space organized as a tree
- Depth-first search: pick always the first Γ-false axiom from the list, and use the 'simplest' ('oldest') instance
- Obviously incomplete, but often OK with favourable ordering of coherent axioms:
  1. Facts first, then Horn clauses ($\rightarrow$ goal first)
  2. Disjunctive clauses (cause branching)
  3. Existential axioms (cause new variables)
  4. Disjunctive existential axioms (cause both, the worst)
- Example: $\exists y.\, s(x, y)$ should never be put first!

```
% the diamond property is preserved under reflexive closure

name(dpe).  :- dynamic e/2,r/2,re/2.

% domain elements a,b,c
dom(a). dom(b). dom(c).

_ axiom assump : (true => re(a,b),re(a,c)).
_ axiom goal_ax(X): (re(b,X),re(c,X) => goal).

% equality axioms, insofar needed
_ axiom ref_e(X) :(dom(X) => e(X,X)).
_ axiom sym_e(X,Y) :(e(X,Y) => e(Y,X)).
_ axiom congl(X,Y,Z) :(e(X,Y),re(Y,Z) => re(X,Z)).

% intro and elim axioms for re
_ axiom e_in_re(X,Y) :(e(X,Y) => re(X,Y)).
_ axiom r_in_re(X,Y) :(r(X,Y) => re(X,Y)).
_ axiom e_or_r(X,Y) :(re(X,Y) => e(X,Y);r(X,Y)).

_ axiom dp(X,Y,Z) :(r(X,Y),r(X,Z) => dom(U),r(Y,U),r(Z,U)).
```

Breadth-first proof search in CL

- Recall: Γ is the condition of the leaf node at hand
- Breadth-first search: collect all 'simplest' instances of Γ-false axioms and use them exhaustively
- Breadth-first search: complete, but often infeasible
- With only constants, depth-first complete for forms 1 and 2
- Depth-first search not complete for one single existential clause, subtle: `p(a).  p(b).  q(b) -> goal.`
  `p(X),p(Y) -> dom(U),p(U),q(X),r(Y).`
- Wanted: fair application of axioms of form 3 and 4 (sl. 21)
- Cycling depth-first: depth-first for forms 1 and 2, plus cycling through the (disjunctive) existential clauses, using instances with the 'oldest' constants first. Complete.

Automated reasoning in CL, conclusions

- Good start: Newman's Lemma (Bezem & Coquand,'03)
- Limited success in CASC: 50% in FOF (Geo, Nivelle'06)
- Readable proofs can be extracted from CL proofs
- Highlight: Hessenberg's Theorem (B, Hendriks, JAR'08)
- Promising: using SAT techniques (Janicic et al.)
- A case study, if time allows: Newman's Lemma, stating that, for any strongly terminating relation $r(x, y)$, if r is locally confluent, then r is confluent. Informal proof on blackboard, code in `nl.in`. Many interesting aspects.

Proof assistants

- In proof assistants, proof objects are required
- CL proofs are readable and easily convertable
- Provers outputting proof objects:
  - `cl.pl` (B, exports proofs to Coq, also used to verify them)
  - `coherent` (Isabelle tactic, Berghofer)
  - ArgoCLP (Coq, Isar, natural language)
- Modern automated support of proof assistants centers around specialized tools for decidable fragments of FOL, using SAT Modulo Theories-techniques. Very useful is, e.g., the tactic `lia` (linear arithmetic) in Coq.

Model finding

- Satisfiability in FOL is co-RE, so restrict to finite models
- Naive approach: try to find a model with 1 element, then with 2 elements, and so on. Quantifiers $\forall, \exists$ are written out ('grounding'), and the resulting (rapidly growing) propositions are fed to a SAT-solver
- Many clever tricks can actually make this to work
- CL proof search is not finite model complete: $\exists y.\ s(x, y)$
- Solution (Nivelle): use (exhaustively) old constants before you generate a new one + use lemma learning
- Success in CASC'07: 81% in FNT (Geo, Nivelle) (Paradox, based on Minisat, winner with 85%)
- CL competitive on problems 'too big to ground'

Constructive algebra

- Pioneers of applying CL/GL to constructve algebra: Coste, Lombardi, Roy, Coquand
- Idea: making constructive sense of classical proofs by exploiting that significant parts of algebra can be formalized in CL/GL
- Barr's Theorem guarantees then that classical results are provable in CL/GL

Algebraic theories in CL/GL

- Ring (commutative with $1 \neq 0$): equational axioms
- Local ring: $\exists y.\ (x \cdot y = 1) \vee \exists y.\ ((1 - x) \cdot y = 1)$
- Field: $(x = 0) \vee \exists y.\ (x \cdot y = 1)$ (makes $=$ decidable!)
- Alg. closed: $\exists x.\ x^{n+1} = a_0 + a_1 x + \cdots + a_n x^n$ (all $n \in \mathbb{N}$), so infinitely many coherent axioms
- Positive formula using an infinite disjunction: $\bigvee_{n \in \mathbb{N}} 0 = x^{n+1}$, expressing that x is nilpotent

Hilbert's Nullstellensatz

- Consider fields $k \subset K$ with K algebraically closed. Let I be an ideal of $k[\vec{x}]$, and $V(I)$ the set of common zeros (Nullstellen) in K of the polynomials in I. Then: for any $p \in k[\vec{x}]$ such that p is zero on $V(I)$ there exists an n such that $p^n \in I$.
- Example: $\mathbb{Q} \subset \mathbb{C}$, $I = (x^4 + 2x^2 + 1)$, $p = x^5 - x$, $p^2 \in I$
- In its full generality, Hilbert's Nullstellensatz is a strong classical theorem, with lots of special cases and variations
- Effective Nullstellensatz: compute the n such that $p^n \in I$
- Dynamical method in algebra: Effective Nullstellensätze, Coste, Lombardi, Roy, 2001 (Dynamic method = CL proof)