Games with Sequential Backtracking and Complete Game Semantics for Intuitionistic, EM-1, and Classical Arithmetic

Workshop on Logical Dialogue games
Thursday, June 29, 2015, Wien

Stefano Berardi C.S. Dept., Turin University,
http://www.di.unito.it/~stefano
Makoto Tatsuta, National Institute of Informatics, Tokyo
http://research.nii.ac.jp/~tatsuta/
Abstract of the Talk

1. Starting from any game with possibly turn conflict, we add the rule of **Sequential Backtracking** for one player.

2. If we start from Tarski games, we obtain a sound and complete game semantics for IPA⁻, Arithmetic with implication as a primitive connective and EM⁻¹, Excluded Middle restricted to 1-quantifier formulas.

3. There is a tree isomorphism (a kind of "Curry Howard" isomorphism) between: proofs of IPA⁻, expressed by an infinitary sequent calculus, and the winning strategies for games with sequential backtracking. We may "run" proofs as game strategies.

4. This isomorphism interprets arithmetical sub-classical proofs as programs **which learn by trials and errors**. These results extend to Intuitionistic and Classical Arithmetic.
Comparing with Polarized Games

1. We produce a complete model for EM-1. **There is no obvious way to restrict Polarized games** in order to give a complete semantics of EM-1.

2. Polarized games give a complete game theoretical model of **provability in Classical logic**. We produce a complete model of **truth for full Classical Arithmetic**.

3. In Polarized games, $\lambda\mu$-terms are in one-to-one with recursive winning strategies. In our game semantics, $\lambda\mu$-terms representing different classical **proofs** may be interpreted by the same recursive winning strategy.

4. Our interpretation produces **a simplified representation of the classical proof as programs**, focused on input/output behavior, on the way the stack of previous states is used, and skipping all the rest.
§1. Games with turn conflicts

• There are two players, $\mathcal{E}$ (Eloise) and $\mathcal{A}$ (Abelard).

• The set of rules for a game $G$ with turn conflicts is a tree with nodes and edges having the color either of $\mathcal{E}$ or of $\mathcal{A}$. Nodes are positions of the game, edges are moves.

• The play starts at the root of $G$. At each turn, a player may: either drop out and lose the game, or move from the current node along an edge of his color, or wait for his opponent’s move.

• If both $\mathcal{E}$ or $\mathcal{A}$ want to move, or both want to wait, we say there is a turn conflict. In this case, the player having the color the node succumbs, and must change its choice.
An example of turn conflict

Both $\mathcal{E}$ or $\mathcal{A}$ may move from a node having the color of $\mathcal{A}$. If both want to move, $\mathcal{A}$ waits and $\mathcal{E}$ moves. If both want to wait, $\mathcal{A}$ moves and $\mathcal{E}$ waits. $\mathcal{A}$ is the player having the color the node, the succumbing player, therefore he is forced to change its choice.
Winner of a game

• In any leaf of G there are no moves left for both players: the succumbing player is forced to drop out.
• The player who drops out loses.
• If G is a finite game (all branches of G are finite), we decide in this way the winner for all plays.
• Otherwise there are infinite plays. In this case, G is equipped with two disjoint sets of infinite plays: \( W_\mathcal{E} \) and \( W_\mathcal{A} \).
• \( \mathcal{E} \) wins if the infinite play is in \( W_\mathcal{E} \), and \( \mathcal{A} \) wins if the infinite play is in \( W_\mathcal{A} \). Otherwise both players loses.
When all edges have the same color of the initial node of the edge, we obtain the usual notion of game, **without turn conflicts**.
Adding backtracking simplifies strategies

- Winning strategy for a game $G$ are often non-recursive, even when $G$ is a recursive tree.

- If we allow $E$ to retract finitely many times her move, many winning strategies for $E$ become recursive. In fact, winning strategies for $E$ become programs learning the correct move by trial and error.

- We may extend any game $G$ with conflict with the possibility for $E$ of retracting any previous move.

- This notion of game is new: we call it $G$ with Sequential Backtracking or $\text{Seq}(G)$. $\text{Seq}(G)$ always has turn conflicts, even if $G$ had no conflicts.
A new notion of game: Seq(G)

- The **color of a node** in Seq(G) is the same as in G.
- The moves of $\mathcal{A}$ in Seq(G) and in G are the same.
- $E$ may move **from any position** in Seq(G) (of any color), and has two kinds of possible moves.
  1. **Explicit Backtracking.** $E$ may come back to any previous node in the history of the play, then $E$ duplicates it as next move
  2. **Implicit Backtracking.** $E$ may come back to any previous node in the history of the play from which $E$ may move, then $E$ produces a move in the original G from it as next move.
The winner of an infinite play in Seq(G)

• We include here the winning condition for infinite plays of Seq(G) only in the case G is a finite play. In this case we ask: all infinite plays in Seq(G) are won by A.

• Why? In Seq(G), \(E\) is allowed to retract finitely many times her previous move, but only in order to find a better move by trial-and-error.

• If G is a finite play, a play in Seq(G) is infinite only if \(E\) changes infinitely many times her move from a given node, just to waste time and to avoid losing the game.

• This behavior is unfair and therefore is penalized: \(E\) loses any infinite play.
Adding Sequential Backtracking to Tarski games

• We define $\text{Classical}(A) = \text{Seq}(\text{Tarski}(A))$ the game obtained adding sequential backtracking to the Tarski game for A.

• Theorem (Completeness for Tarski games with seq. back.). $E$ has a winning strategy for $\text{Classical}(A)$ if and only if $E$ has a recursive winning strategy for $\text{Classical}(A)$ if and only if $A$ is true.

• Adding backtracking does not change the winner, but makes the winning strategy recursive. The winning strategy is now a program learning the winning moves by trial-and-error. Any wrong move of $E$ may be changed, provided we find the right one in finite time.
§2. Proofs as programs which learn.

• In Classical(A), classical proofs of A are interpreted as programs learning the value of a witness for an existential statement by trial-and-error. This is possible even when no program computing the witness exists. We include a toy example with primitive implication (this is new).

• Assume P is any recursive predicate such that the predicate $\exists y. P(x,y)$ is not recursive. We claim that $E$ has a winning strategy from the judgement:

$$true.EM_1 = \text{true.} \forall x. ( \exists y. P(x,y) \rightarrow \bot \lor \exists y. P(x,y) )$$

but $E$ has no recursive winning strategy, unless we allow backtracking.
A non-recursive winning strategy for Tarski(EM₁)

\( \mathcal{A} \) moves: \( \text{true.}\forall x. (\exists y. P(x,y) \rightarrow \bot \lor \exists y. P(x,y)) \)

\( \mathcal{E} \) moves: \( \text{true.} \exists y. P(a,y) \rightarrow \bot \lor \exists y. P(a,y) \)

If \( P(a,b) \) is true, then \( \text{true.} P(a,b) \) is conjunctive, with the color of \( \mathcal{A} \). \( \mathcal{A} \) should move, he cannot and he drops out.
A recursive winning strategy for \text{Classical} (\text{EM}_1)

$A$ moves: \quad true. \forall x. (\exists y. P(x,y) \rightarrow \bot \lor \exists y. P(x,y))$

$E$ moves: \quad true. \exists y. P(a,y) \rightarrow \bot \lor \exists y. P(a,y)$

$E$ moves: \quad true. \exists y. P(a,y) \rightarrow \bot$

$A$ moves: \quad false. \exists y. P(a,y)$

... \quad false. P(a,b) ... \quad ... \quad true. P(a,b) ...

If $P(a,b)$ is true, then \textit{false}. $P(a,b)$ is disjunctive, with the color of $E$. $E$ cannot choose a child of \textit{false}.$P(a,b)$. Thus, $E$ backtracks, then $E$ chooses $P(a,b)$, which is true, and wins.
Implementing a restricted form of Backtracking

• There is a restriction of backtracking we call $EM_1$-backtracking, in which whenever some positive formulas are discarded from the history of the play, they are never restored.

• Theorem (Completeness of $EM_1$-backtracking) $EM_1$-backtracking validates exactly the theorems of $IPA^-$ (formulas with implication which are intuitionistic consequences of $EM_1$ and of recursive $\omega$-rule).

• The interest of this result lies in the possibility of "running" some classical proofs using less memory space and less memory structure, therefore less time.

• If we restrict backtracking to a positive formula to the last positive formula, then we obtain Intuit. Arithmetic + $\omega$-rule.
Conclusion

• The proof/strategy isomorphism provides a way of describing classical proofs as programs which learn, alternative to Griffin’s use of continuations.

• With respect to the original isomorphism proposed by H. Herbelin, we added implication as primitive connective.

• The challenge is now to provide some implementation of proofs suggested by this new way of looking at proofs.

• The study of game semantics may provide further information: if we have a proof with a limited use of classical logic (say, using EM₁-logic), its interpretation as strategy makes a limited use of backtracking, therefore it has a simpler implementation.

• Differently from Polarized games, our interpretation cannot be used to represents the λμ-formulation of classical proofs.
Index

• §1. Games with conflicts.
• §2. Proofs as programs which learn.
• Appendix 1. A definition of Tarski games over judgements.
• Appendix 2. A formulation of Classical Arithmetic $\text{PA + \omega-rule}$ satisfying the proof/strategy isomorphism (for proofs in a simplified form)
Appendix 1. Tarski games over judgements

• Tarski games are the canonical notion of games (without turn conflicts) representing the truth of an arithmetical statement. In order to define Tarski games, we consider a first order language $L$: $\text{True}, \text{False}, \lor, \land, \neg, \to, \forall, \exists$, with all primitive recursive predicates and functions.

• We define a relation $<_1$ (immediate subformula) for closed formulas of $L$. We set $A <_1 \neg A$ and:

\[
A, B <_1 A \lor B, A \land B, A \to B
\]

\[
A[t/x] <_1 \forall x. A, \exists x. A \quad (\text{for all closed terms } t)
\]
Disjunctive, conjunctive, positive and negative formulas

- \( A \lor B, \ \exists x. A, \ A \rightarrow B, \ \neg A \) are **disjunctive** formulas.
- \( A \land B, \ \forall x. A \) are **conjunctive** formulas.
- \( A <_1 A \rightarrow B, \ \neg A \) is a **negative** subformula. In all other cases \( A <_1 C \) is a **positive** subformula.
- Disjunctive formulas correspond to **sending an output** (to the outside), conjunctive formula to **receiving an input** (from the outside).
- Negative formulas correspond to **questions** (both from us and from outside) and positive formulas to **answers** (both from us and from outside).
Disjunctive, conjunctive, positive and negative “judgements”

- **Judgements**: \( J = s.A \), where either \( s=\text{true} \) or \( s=\text{false} \).
- **true.\( A \)** is a positive judgement. **true.\( A \)** is disjunctive (conjunctive) iff \( A \) disjunctive (conjunctive).
- **false.\( A \)** is a negative judgement. **false.\( A \)** is disjunctive (conjunctive) iff \( A \) conjunctive (disjunctive).
- \( s.\( A <_1 t.\( B \) \) if and only if: \( A <_1 B \), and \( s=t \) if \( A \) is a positive subformula of \( B \), and \( s \neq t \) if \( A \) is a negative subformula.
- For instance, **false.\( A \), true.\( B <_1 \) true.\( A \rightarrow \)B**.
- We write a conjunctive judgement \( J \) as \( \land_{i \in I} J_i \) for all \( J_i <_1 J \), and a disjunctive judgement \( J \) as \( \lor_{i \in I} J_i \) for all \( J_i <_1 J \).
The game Tarski(s.A)

• We write $\leq$ for the transitive closure of $<_1$. For each judgement s.A we define Tarski(s.A), the game associated to the notion of truth for s.A. We write Tarski(A) for Tarski(true.A).

• The nodes of Tarski(s.A) are all judgements t.B $\leq$ s.A. The root is s.A, the child/father relation is t.B $<_1$ u.C.

• Disjunctive formulas and edges from them are colored $E$, conjunctive formulas and edges from them are colored $A$.

• Theorem (Completeness for Tarski games and Truth). $E$ has an arithmetical winning strategy from Tarski(A) if and only if A is true. The strategy selects a true immediate subjudgement if any exists.
Appendix 2. A formulation of PA+ω-rule with the proof/strategy isomorphism

• The language of PA+ω-rule are all judgements. Any judgement is of the form $\bigvee_{i \in I} J_i$ or $\bigwedge_{i \in I} J_i$. Say: $\text{true.}A \to B = \bigvee\{\text{false.}A, \text{true.}B\}$ and $\text{false.}A \to B = \bigwedge\{\text{true.}A, \text{false.}B\}$.

• Sequents of CLω are ordered lists of judgements. Therefore Contraction and Exchange rules are not built-in in the notion of sequent.

• We explicitly assume Contraction in PA+ω-rule. We hyde Exchange rule through the fact that the active formula, if disjunctive, may be in any position in the sequent.

• Identity rule is trivially derivable in PA+ω-rule. Cut rule is derivable as well, but highly non-trivial.
A formulation of $\text{PA}+\omega$-rule with 3 rules
(in one-side form, with judgements)

\[ \Gamma, \forall_{i \in I} J_i, \Delta, J_i \quad (\text{disj. with implicit contraction and exchange: for some } i \in I) \]

\[ \Gamma, \forall_{i \in I} J_i, \Delta \]

\[ \Gamma, \land_{i \in I} J_i, J_i \quad (\text{all } i \in I) \quad (\text{conj. with implicit contr.: for all } i \in I, \text{ and recursively in } i) \]

\[ \Gamma, \land_{i \in I} J_i \]

Remark the asymmetry with $\forall$: we do not have $\Gamma, \land_{i \in I} J, \Delta$

\[ \Gamma, J, \Delta, J \quad (\text{contraction with implicit exchange}) \]

\[ \Gamma, J, \Delta \]
Proof/Strategy Isomorphism and Cut-Elimination Theorem

**Theorem.** Let A be any closed arithmetical formula.

1. **(Soundness and Completeness)** A formula A is a theorem of \( \text{PA} + \omega \)-rule if and only if \( \mathcal{E} \) has a recursive winning strategies on the game Classical(true.A).

2. **(Curry-Howard)** The recursive winning strategy-trees for \( \mathcal{E} \) on Classical(true.A) are tree-isomorphic to the infinitary recursive cut-free proof-trees of A in \( \text{PA} + \omega \)-rule.

3. **(Cut-Elimination)** It is translated in a game-theoretical result: “any dialogue between two terminating strategies for \( \mathcal{E} \) on Classical(true.A) and Classical(false.A) is terminating”.

24
Bibliography

[As1] F. Aschieri. Learning Based Realizability for HA + EM1 and 1-Backtracking Games: Soundness and Completeness. To appear on APAL.


Bibliography


Bibliography

[Laur1]
http://dblp.uni-trier.de/db/journals/apal/apal130.html#Laurent04

[Laur2]
O. Laurent: *Game semantics for first-order logic*. Logical Methods in Computer Science 6(4)