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.

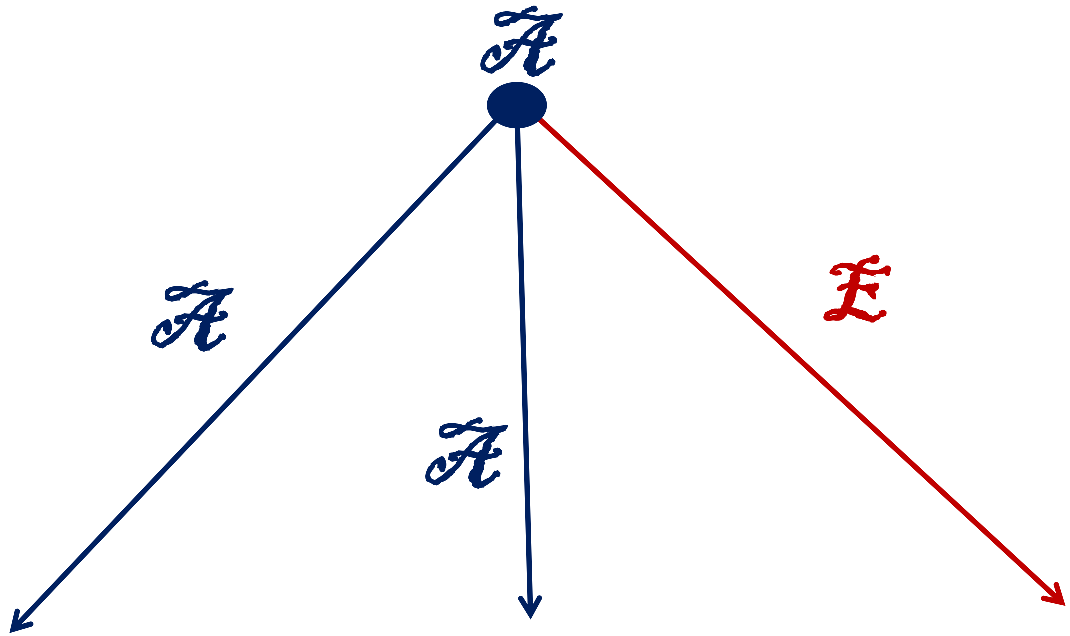
Comparing with Polarized Games

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, λμ-terms are in one-to-one with recursive winning strategies. In our game semantics, **λμ-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, $\mathfrak{E}$ **(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 $\mathfrak{E}$ 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** $\mathfrak{E}$ 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

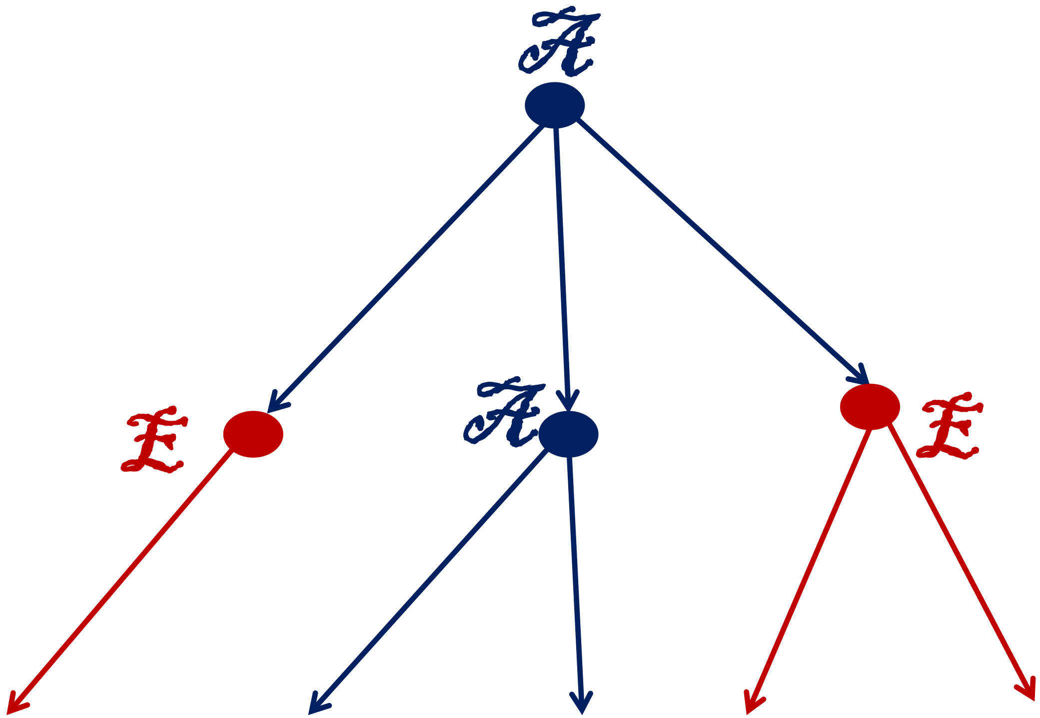


Both $\mathcal{E}$ or $\mathcal{A}$ may move from a node having the color of $\mathcal{A}$. If both want to move, $\mathcal{A}$ waits and $\mathcal{E}$ moves. If both want to wait, $\mathcal{A}$ moves and $\mathcal{E}$ waits. $\mathcal{A}$ is the player having the color the node, the **succumbing player**, 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: $\mathbf{W}_{\mathcal{E}}$ and $\mathbf{W}_{\mathcal{A}}$.
- $\mathcal{E}$ wins if the infinite play is in $\mathbf{W}_{\mathcal{E}}$, and $\mathcal{A}$ wins if the infinite play is in $\mathbf{W}_{\mathcal{A}}$. Otherwise both players loses.

Games without turn conflict



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 $\mathfrak{E}$ to **retract finitely many times** her move, many winning strategies for $\mathfrak{E}$ become **recursive**. In fact, winning strategies for $\mathfrak{E}$ become programs learning the correct move **by trial and error**.
- We may extend any game **G** with conflict with the possibility for $\mathfrak{E}$ of retracting any previous move.
- This notion of game is **new**: we call it **G with Sequential Backtracking** or **Seq(G).** 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.
- $\mathfrak{E}$ may move **from any position** in Seq(G) (of any color), and has two kinds of possible moves.

1. ***Explicit Backtracking.*** $\mathfrak{E}$ may come back to any previous node in the history of the play, then $\mathfrak{E}$ **duplicates** it as next move
2. ***Implicit Backtracking.*** $\mathfrak{E}$ may come back to any previous node in the history of the play from which $\mathfrak{E}$ may move, then $\mathfrak{E}$ 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** $\mathcal{A}$.
- Why? In Seq(G), $\mathfrak{E}$ 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 $\mathfrak{E}$ **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**: $\mathfrak{E}$ loses any infinite play.

Adding Sequential Backtracking to Tarski games

- We define **Classical(A)=Seq(Tarski(A))** the game obtained adding sequential backtracking to the Tarski game for A.
- **Theorem (Completeness for Tarski games with seq. back.).** $\mathfrak{E}$ has a winning strategy for Classical(A) if and only if $\mathfrak{E}$ has a **recursive winning** strategy for 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 $\mathfrak{E}$ 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 interpred 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 ∃y.P(x,y) is <u>not</u> recursive. We claim that $\mathfrak{E}$ has a winning strategy from the judgement:

  true.EM_1 = **true.∀x.(∃y.P(x,y)→⊥ ∨ ∃y.P(x,y))**

  but $\mathfrak{E}$ has no recursive winning strategy, unless we allow backtracking.

A non-recursive winning strategy for Tarski(EM_1)

$\mathcal{A}$ **moves:** true.$\forall$x. ($\exists$y.P(x,y)$\rightarrow\bot\vee\exists$y.P(x,y))

$\mathcal{E}$ **moves:** true.$\exists$y.P(a,y)$\rightarrow\bot \vee \exists$y.P(a,y)

true.$\exists$y.P(a,y)$\rightarrow\bot$

$\mathcal{E}$ **moves:** true.$\exists$y.P(a,y)

false.$\exists$y.P(a,y)

… false.P(a,b) … … true.P(a,b) …

If P(a,b) is true, then 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 Classical(EM_1)

$\mathcal{A}$ **moves:** **true.∀x. (∃y.P(x,y)→⊥∨∃y.P(x,y))**

$\mathcal{E}$ **moves:** **true.∃y.P(a,y)→⊥ ∨ ∃y.P(a,y)**

$\mathcal{E}$ **moves: true.∃y.P(a,y)→⊥**

true.∃y.P(a,y)

$\mathcal{A}$ **moves: false.∃y.P(a,y)**

… **false.P(a,b)** … … **true.P(a,b)** …

If P(a,b) is true, then **false. P(a,b)** is disjunctive, with the **color of $\mathcal{E}$**. $\mathcal{E}$ cannot choose a child of **false.P(a,b)**. Thus, $\mathcal{E}$ backtracks, then $\mathcal{E}$ 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 ω-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 + ω-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 **EM_1-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 $\lambda\mu$-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 **PA + ω-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 (<u>without</u> turn conflicts) representing the truth of an arithmetical statement. In order to define Tarski games, we consider a first order language **L: True, False, $\vee$, $\wedge$, $\neg$, $\rightarrow$, $\forall$, $\exists$**, with all primitive recursive predicates and functions.
- We define a relation $<_1$ (immediate subformula) for <u>closed</u> formulas of **L**. We set $A <_1 \neg A$ and:

$$A, B \quad <_1 \quad A \vee B, \quad A \wedge B, \quad A \rightarrow B$$

$$A[t/x] \quad <_1 \quad \forall x.A, \exists x.A \qquad \textit{(for all closed terms t)}$$

Disjunctive, conjunctive, positive and negative formulas

- $A \vee B$, $\exists x.A$, $A \rightarrow B$, $\neg A$ are **disjunctive** formulas.
- $A \wedge B$, $\forall x.A$ 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=true or s=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≠t if A is a negative subformula.
- For instance, **false.A, true.B $<_1$ true.A→B**.
- We write a conjunctive judgement **J** as $\wedge_{i \in I} J_i$ for all $J_i <_1 J$, and a disjunctive judgement **J** as $\vee_{i \in I} J_i$ 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** $\mathscr{E}$, conjunctive formulas and edges from them are **colored** $\mathscr{A}$.
- **Theorem (Completeness for Tarski games and Truth).** $\mathscr{E}$ 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 $\vee_{i\in I}J_i$ or $\wedge_{i\in I}J_i$. Say: **true.A→B = ∨{false.A,true.B}** and **false.A→B=∧{true.A, false.B}**.
- Sequents of **CL_ω** 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 PA+ω-rule with 3 rules
(in one-side form, with judgements)

$$\frac{\Gamma, \vee_{i\in I} J_i, \Delta, J_i}{\Gamma, \vee_{i\in I} J_i, \Delta}$$
(disj. with implicit contraction and exchange: for some i∈I)

$$\frac{\Gamma, \wedge_{i\in I} J_i, J_i \quad (\text{all } i\in I)}{\Gamma, \wedge_{i\in I} J_i}$$
(conj. with implicit contr.: for all i∈I, and <u>recursively</u> in i)

Remark the asymmetry with ∨: we do <u>not</u> have $\Gamma, \wedge_{i\in I} J, \Delta$

$$\frac{\Gamma, J, \Delta, J}{\Gamma, J, \Delta}$$
(contraction with implicit exchange)

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 **PA+ω-rule** if and only if $\mathfrak{E}$ has a recursive winning strategies on the game Classical(true.A).
2. ***(Curry-Howard)*** The recursive winning strategy-trees for $\mathfrak{E}$ on Classical(true.A) are tree-isomorphic to the infinitary recursive cut-free proof-trees of A in **PA+ω-rule**.
3. ***(Cut-Elimination)*** It is translated in a game-theoretical result: "any dialogue between two terminating strategies for $\mathfrak{E}$ on Classical(true.A) and Classical(false.A) is **terminating**".

Bibliography

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

[As2] F. Aschieri. ***Learning, Realizability and Games in Classical Arithmetic.*** Ph. D. thesis, Torino, 2011.

[Be1] S. Berardi, T. Coquand, and S. Hayashi. ***Games with 1-backtracking.*** APAL, 2010.

[Be2] S. Berardi and M. Tatsuta. ***Positive Arithmetic Without Exchange Is a Subclassical Logic.*** In Zhong Shao, editor, APLAS, volume 4807 of Lecture Notes in Computer Science, pages 271-285. Springer, 2007.

Bibliography

[Be3] S. Berardi and Y. Yamagata. ***A Sequent Calculus for Limit ComputableMathematics.*** APAL, 153(1-3):111-126, 2008.

[Coq] T. Coquand. ***A Semantics of Evidence for Classical Arithmetic.*** JSL, 60(1):325-337, 1995.

[Fel] W. Felscher. ***Dialogues as a foundation for intuitionistic logic.*** In D.M. Gabbay and F. Guenthner, editors, Handbook of Philosophical Logic. Vol. III, pages 341–372. Dordrecht: D. Reidel, 1986.

[Her] Hugo Herbelin. ***A Lambda-Calculus Structure Isomorphic to Gentzen-Style Sequent Calculus Structure***. CSL 1994: 61-75

Bibliography

[Laur1]

O. Laurent: ***Polarized games***. Ann. Pure Appl. Logic 130(1-3) : 79-123 (2004). In:

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)