Structural Analysis of Narratives with Coq

ITP 2011

Anne-Gwenn Bosser[1] Julien Forest[2]
Marc Cavazza[1] *Pierre Courtieu*[3]

[1]University of Teesside, School of Computing
[2]École nat. sup. d'informatique pour l'industrie et l'entreprise, CEDRIC
[3]Conservatoire National des Arts et Métiers, CEDRIC

Motivations

- Story telling, Video games
- Game designers
- “Quality of experience”
- *Emma Bovary*

Why Formalizing?

- As usual: Verify properties of (interactive) narratives:

  Until user choses between A and B he can still win.

  (less usual)*Reaching end state* $\mathcal{G}$ *from resources* $\mathcal{R}$ *and actions* $\mathcal{A}$ *makes a choice between A and B mandatory*.

- Unusual: Help building the formalism.

Some non formal remarks

Plot

Actions linked by causality, resource consumption and initial conditions.

Narrative

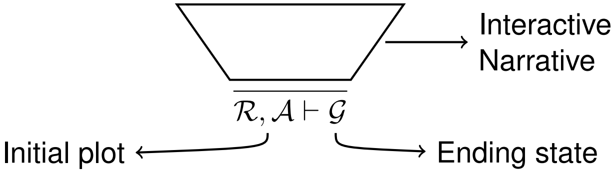
Effective “story” following a plot. Temporal.

Interactive Narrative

Set of *narratives* sharing a same plot. Tree-like, nodes correspond to *user choices/interactions*. . .

Causal + Temporal + Resources

Narrative $\equiv$ ILL proof of the plot[4]



- $\mathcal{R}$: Resources, Initial Conditions
- $\mathcal{G}$: Goal (available resources at end)
- $\mathcal{A}$: Narrative Actions ($\Phi \multimap \Psi$)

[4] Bosser, Cavazza, Champagnat. *Linear logic for non-linear storytelling*. ECAI'2010/FAIA.

Plot I: Narrative Resources (Atoms, $\otimes$)

Examples (Atoms)

"Emma Bovary dies"	D
"Emma Bovary saves herself"	A
"Emma Bovary accepts Guillaumin's proposition"	AG

- Can be "consumed", i.e. "happen"
- Grouped with $\otimes$

Plot II: Narrative Actions ($\multimap$)

- Takes resources and returns actions and resources

Example

A produces action "*B*" when it happens $A \multimap B$
B produces nothing when it happens $B \multimap 1$

- Apply $\multimap_L$ rule $\equiv$ make an action happen

$$\multimap_L \frac{Y, \Gamma \vdash \Delta}{X \multimap Y, X, \Gamma \vdash \Delta}$$

$$\cfrac{\cfrac{!X, Y, \Gamma \vdash \Delta}{X \multimap Y, X, !X, \Gamma \vdash \Delta}\ !_I}{X \multimap Y, !X, \Gamma \vdash \Delta}\ \multimap_L$$

$\oplus$ (external) choice

Example

User can chose between A and B	$A \oplus B$
User can chose to do C or not	$C \oplus 1$

& (internal) choice

Example

Guillaumin can chose between A or B	$A \,\&\, B$
C may happen or not	$C \,\&\, 1$

Difference between & and $\oplus$ – Left rules

$$\dfrac{\dfrac{A, \Gamma \vdash \Delta}{A \,\&\, B, \Gamma \vdash \Delta}\&_{L_1} \quad \dfrac{C, \Gamma \vdash \Delta}{A \,\&\, C, \Gamma \vdash \Delta}\&_{L_2}}{(A \,\&\, B) \oplus (A \,\&\, C), \Gamma \vdash \Delta}\oplus_L \qquad \dfrac{\dfrac{A, \Gamma \vdash \Delta}{A \,\&\, B, \Gamma \vdash \Delta}\&_{L_1} \quad \dfrac{A, \Gamma \vdash \Delta}{A \,\&\, C, \Gamma \vdash \Delta}\&_{L_1}}{(A \,\&\, B) \oplus (A \,\&\, C), \Gamma \vdash \Delta}\oplus_L$$

- $\oplus$: External choice (open world assumption)
- Internal choice: specifies one particular story (interactive narrative), may depend on previous external choices

Restriction on formulas

Initial sequent of a narrative: $\mathcal{R}, \mathcal{A} \vdash \mathcal{G}$

- $\mathcal{R}$: Resources, Initial Conditions
- $\mathcal{G}$: Goal
- $\mathcal{A}$: Narrative Actions

Restriction on formulas

Initial sequent of a narrative: $\mathcal{R}, \mathcal{A} \vdash \mathcal{G}$

- $\mathcal{R}$: Resources, Initial Conditions $1 | \mathit{atom} | \mathcal{R} \,\&\, \mathcal{R} | \mathcal{R} \otimes \mathcal{R} | !\mathcal{R}$
- $\mathcal{G}$: Goal
- $\mathcal{A}$: Narrative Actions

Restriction on formulas

Initial sequent of a narrative: $\mathcal{R}, \mathcal{A} \vdash \mathcal{G}$

- $\mathcal{R}$: Resources, Initial Conditions $\quad 1 | \textit{atom} | \mathcal{R} \,\&\, \mathcal{R} | \mathcal{R} \otimes \mathcal{R} | !\mathcal{R}$
- $\mathcal{G}$: Goal $\quad 1 | \textit{atom} | \mathcal{G} \,\&\, \mathcal{G} | \mathcal{G} \otimes \mathcal{G} | \mathcal{G} \oplus \mathcal{G}$
- $\mathcal{A}$: Narrative Actions

Restriction on formulas

Initial sequent of a narrative: $\mathcal{R}, \mathcal{A} \vdash \mathcal{G}$

- $\mathcal{R}$: Resources, Initial Conditions $\quad 1|atom|\mathcal{R} \,\&\, \mathcal{R}|\mathcal{R} \otimes \mathcal{R}|!\mathcal{R}$
- $\mathcal{G}$: Goal $\quad 1|atom|\mathcal{G} \,\&\, \mathcal{G}|\mathcal{G} \otimes \mathcal{G}|\mathcal{G} \oplus \mathcal{G}$
- $\mathcal{A}$: Narrative Actions $\quad 1|\mathcal{R}_\mathcal{A} \multimap \mathcal{C}|\mathcal{A} \,\&\, \mathcal{A}|\mathcal{A} \oplus \mathcal{A}|!\mathcal{A}$
- $\mathcal{R}_\mathcal{A}$: Consumable resources $\quad 1|atom|\mathcal{R}_\mathcal{A} \otimes \mathcal{R}_\mathcal{A}$
- $\mathcal{C}$: Intermediate context $\quad \mathcal{R}|\mathcal{A}|\mathcal{C} \otimes \mathcal{C}$

Theorem (Coq)

$\mathcal{C} \vdash \mathcal{G}$ stable by application of rules (no cut)

Coq ILL formalization

```
Inductive ILL_proof: env→ formula→ Prop:=
|Id: ∀ Γ p, Γ == {p}→ Γ⊢p
|Impl_R: ∀ Γ p q, p::Γ⊢q→  Γ⊢p ⊸ q
|Impl_L: ∀ Γ Δ Δ' p q r, (p ⊸ q) ∈Γ→(Γ\(p ⊸ q))== Δ ∪ Δ'→ Δ⊢p→ q:: Δ' ⊢r  → Γ⊢r
|Times_R: ∀ Γ Δ Δ' p q, Γ == Δ ∪ Δ'→  Δ⊢p  → Δ' ⊢q  → Γ⊢p ⊗ q
|Times_L: ∀ Γ p q r, (p ⊗ q) ∈Γ→ q::p::(Γ\(p ⊗ q))⊢r  → Γ⊢r
|One_R: ∀ Γ, Γ == ∅  → Γ⊢1
|One_L: ∀ Γ p, 1∈Γ  → (Γ\1)⊢p  → Γ⊢p
|And_R: ∀ Γ p q, Γ⊢p  → Γ⊢q  → Γ⊢(p & q)
|And_L_1: ∀ Γ p q r, (p & q) ∈Γ→ p::(Γ\(p&q))⊢r  → Γ⊢r
|And_L_2: ∀ Γ p q r, (p & q) ∈Γ→ q::(Γ\(p&q))⊢r  → Γ⊢r
|Oplus_L: ∀ Γ p q r, (p⊕q) ∈Γ→ p::(Γ\(p⊕q))⊢r→ q::(Γ\(p⊕q))⊢r→ Γ⊢r
|Oplus_R_1: ∀ Γ p q, Γ⊢p  → Γ⊢p⊕q
|Oplus_R_2: ∀ Γ p q, Γ⊢q  → Γ⊢p⊕q
|T_: ∀ Γ, Γ⊢⊤
|Zero_: ∀ Γ p, 0∈Γ  → Γ⊢p
|Bang_D: ∀ Γ p q, !p∈Γ→ p::(Γ\(!p))⊢q  → Γ⊢q
|Bang_C: ∀ Γ p q, !p∈Γ→ !p::Γ⊢q  → Γ⊢q
|Bang_W: ∀ Γ p q, !p∈Γ→ Γ\(!p)⊢q  → Γ⊢q
where " x⊢y ":= (ILL_proof x y).
```

Finding $h : \mathcal{R}, \mathcal{A} \vdash \mathcal{G}$, Dedicated tactics

```
Lemma original :
 { P&1, R, G, B&1, !(S ⊸ A), (E ⊸ A)&1, (P ⊸ D)&1,
  (R ⊸ 1)&(R ⊸ E), (G ⊸ 1)⊕(G ⊸ S), 1⊕((B ⊸ S)&(B ⊸ 1)) } ⊢  A⊕D.
Proof.
```

- Environment management, (110 line Ltac)

```
oplus_L (G ⊸ 1) (G ⊸ S).
```

$$\frac{G \multimap 1, \Gamma \vdash \Delta \qquad G \multimap S, \Gamma \vdash \Delta}{(G \multimap 1) \oplus (G \multimap S), \Gamma \vdash \Delta} \oplus_L$$

- Partial automation of rules (120 lines)

```
search_goal 8 ({ ... } ⊢  A⊕D).
```

Goal Driven, Depth-bound
Non branching

```
finish_goal.
```

Brute force

Checking *P h* by dependently typed programming

```
Program Fixpoint exist (test) `(h: e ⊢ f) {struct h}: boolP :=
  match h with
   | Id _ _ _ ⇒ test h
   | Impl_R _ _ _ x ⇒ test h ORP exist test x
   | Oplus_L _ _ _ _ _ x x0 ⇒ test h ORP exist test x ORP exist test x0
   ...

Program Fixpoint for_all (test) `(h: e ⊢ f) {struct h}: boolP :=
  match h with
   | Id _ _ _ ⇒ test h
   | Impl_R _ _ _ x ⇒ test h ORP for_all test x
   | Oplus_L _ _ _ _ _ x x0 ⇒ test h ORP (for_all test x ANDP for_all test x0)
   ...
```

Limitations:

- **Prop** → **bool** impossible (typing)
- **Type** → **bool** not allowed (Program)
- **Prop** → boolP OK with boolP:**Prop**

Checking *P h* by dependently typed programming

Checking that chosing between `S` and `R` is possible:

```
Definition test_S_R e f `{h: e ⊢ f} :=
  match h with
    | Oplus_R_1 _ p q x ⇒ p ?= S ANDP q ?= R
    | Oplus_R_2 _ p q x ⇒ p ?= S ANDP q ?= R
    | _ ⇒ falseP
  end.

Lemma SR: { G,((B ⊸ S)&(B ⊸ R))&1,(G ⊸ B) ⊕ (G ⊸ S) } ⊢ S⊕R.
Proof.
  oplus_l (G ⊸ B) (G ⊸ S).
  finish_proof.
  finish_proof.
Defined.

Eval compute in (exist test_S_R SR).
```

$\forall h : (\mathcal{R}, \mathcal{A} \vdash \mathcal{G}), P\ h$

- Consider *all* proofs of a sequent
- Taming complexity:
  - Ignore side conditions proofs (ex: $p \multimap q \in \mathcal{G}$), Setoids
  - Cut proof branches with *unprovability lemmas (2300 lines)*
    Some due to the restriction on formulas
  - Adapted inversion tactics (200 lines more of Ltac)
  - Proof sharing

$\forall h : (\mathcal{R}, \mathcal{A} \vdash \mathcal{G}), P\ h$

Lemma $\forall$ h:{G,(B $\multimap$ S&B $\multimap$ R)&1,(G $\multimap$ B) $\oplus$ (G $\multimap$ S)} $\vdash$ S $\oplus$ R, exist test_S_R h = trueP.
Proof.
  one_step. *(dependent inversion + cleaning + unprove)*
  (use (recursively) auxiliary lemmas on each goal (47))
Qed.

Lemma $\forall$ h:{P&1,(V $\multimap$ A)&1,(E $\multimap$ A)&1,(P $\multimap$ M)&1,V} $\vdash$ A $\oplus$ M, exist test_A h = trueP.
Proof.
  one_step.
  (283 auxiliary lemmas)
Qed.

Conclusion

- Formal definition of narratives
  $\rightarrow$ Variance, metrics, normalisation...
- Interesting view on proof terms
  $\rightarrow$ More automation, External procedure?
- Ltac limitations