Relating Apartness and Bisimulation

Herman Geuvers

Radboud Univ. Nijmegen and TU Eindhoven

January 6, 2021

jww Bart Jacobs

Overview

- Bisimulation and apartness
- LTSs and branching bisimulation
- Branching apartness
- Proving properties about branching apartness and using it

Deterministic Finite Automata

A DFA $M = (A, K, \delta, \downarrow)$ consists of an alphabet A, a set of states K and $\delta : K \times A \to K$, $\downarrow: K \to 2$. A DFA M gives rise to the notions of bisimulation for M and apartness for M.

- $R \subseteq K \times K$ is a M-bisimulation if it satisfies the following rule.

$$\frac{R(q_1, q_2)}{q_1 \downarrow \Leftrightarrow \; q_2 \downarrow \quad \wedge \quad \forall a \in A \forall p_1, p_2 (q_1 \to_a p_1 \wedge q_2 \to_a p_2 \implies R(p_1, p_2))}$$

Two states $q_1, q_2 \in K$ are M-bisimilar, $q_1 \underline{\leftrightarrow}^M q_2$, is defined by

$$q_1 \underline{\leftrightarrow}^M q_2 \; := \; \exists R \subseteq K \times K \, (R \text{ is a } M\text{-bisimulation and } R(q_1, q_2)).$$

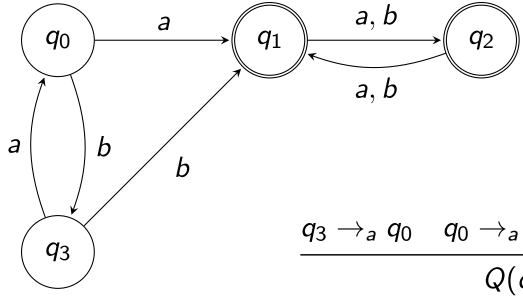
- $Q \subseteq K \times K$ is a M-apartness if it satisfies the following rules.

$$\frac{q_1 \to_a p_1 \qquad q_2 \to_a p_2 \qquad Q(p_1, p_2)}{Q(q_1, q_2)} \qquad\qquad \frac{q_1 \downarrow \not\Leftrightarrow \; q_2 \downarrow}{Q(q_1, q_2)}$$

Two states $q_1, q_2 \in K$ are M-apart, $q_1 \,\#^M\, q_2$, if

$$q_1 \,\underline{\#}^M\, q_2 \; := \; \forall Q \subseteq K \times K \, (\text{if } Q \text{ is a } M\text{-apartness then } Q(q_1, q_2)).$$

Example



$$\dfrac{q_3 \to_a q_0 \quad q_0 \to_a q_1 \quad \dfrac{\neg(q_0 \downarrow) \wedge \ q_1 \downarrow}{Q(q_0, q_1)}}{Q(q_3, q_0)}$$

A bisimulation is given by $q_1 \sim q_2$. It can be shown that $q_0 \mathrel{\underline{\#}}^M q_3$ because for every apartness Q we have the derivation given on the right.

- To be M-apart is the smallest relation satisfying specific closure properties, so it is an inductive property.
- The closure properties yield a proof system for deriving that two elements are M-apart.

For the DFA case, the derivation rules are:

$$\frac{q_1 \to_a p_1 \qquad q_2 \to_a p_2 \qquad p_1 \mathrel{\underline{\#}}^M p_2}{q_1 \mathrel{\underline{\#}}^M q_2} \qquad\qquad \frac{q_1\downarrow \not\Leftrightarrow q_2\downarrow}{q_1 \mathrel{\underline{\#}}^M q_2}$$

Apartness in constructive analysis

In constructive real analysis (and similarly when talking about computable real numbers), one takes apartness as a primitive and defines equality as its negation:

$$x \mathrel{\#} y \;\simeq\; \text{we can find a proper distance } \delta \in \mathbb{Q} \text{ between } x \text{ and } y$$
$$x = y := \neg(x \mathrel{\#} y)$$

A relation is usually only called an apartness relation if it satisfies three properties.

Definition. A relation $\#$ is a proper apartness relation if it is

- irreflexive: $\forall x\, (\neg x \mathrel{\#} x)$,
- symmetric: $\forall x, y\, (x \mathrel{\#} y \implies y \mathrel{\#} x)$,
- co-transitive: $\forall x, y, z\, (x \mathrel{\#} y \implies x \mathrel{\#} z \vee z \mathrel{\#} y)$.

Lemma. For R a relation, R is an equivalence relation if and only if $\neg R$ is a proper apartness relation.

Proof. The only interesting property to check is that R is transitive iff $\neg R$ is co-transitive.

The general categorical picture

Bisimulation and apartness can be defined by induction over the structure of the polynomial functor $F : \mathbf{Set} \to \mathbf{Set}$ that we consider the coalgebra for.

- For DFAs: $c : K \to F(K)$ with $F(X) = X^A \times 2$.
- For streams over A: $c : K \to F(K)$ with $F(X) = A \times X$.

We have the following result relating bisimulation and apartness.
Lemma.

1. R is a c-bisimulation if and only if $\neg R$ is a c-apartness.
2. The relation $\underline{\leftrightarrow}$, the union of all bisimulations: $\underline{\leftrightarrow} = \bigcup\{R \mid R \text{ is a } c\text{-bisimulation}\}$, is itself a c-bisimulation equivalence.
3. The relation $\underline{\#}$, the intersection of all apartness relations: $\underline{\#} = \bigcap\{Q \mid Q \text{ is a } c\text{-apartness relation}\}$, is itself a proper c-apartness relation.
4. $\underline{\leftrightarrow} = \neg\underline{\#}$.

LTSs and branching bisimulation

A labelled transition systems, LTS, is a tuple $(X, A_\tau, \to)$, where

- X is a set of states,
- $A_\tau = A \cup \{\tau\}$ is a set of actions, with τ the special silent action,
- $\to \subseteq X \times A_\tau \times X$ is the transition relation.
  We write $q_1 \to_u q_2$ for $(q_1, u, q_2) \in \to$. Furthermore, $\twoheadrightarrow_\tau$ denotes the reflexive transitive closure of $\to_\tau$.

NB. We reserve $q_1 \to_a q_2$ to denote a transition with an a-step with $a \in A$ (so $a \neq \tau$).

The notion of bisimulation for LTSs we consider is branching bisimulation. Here, the categorical picture is not completely clear, so there is no "canonical" way for constructing the bisimulation and apartness from the functor and the co-algebra.

Branching bisimulation

We give the definition of branching bisimulation in rule style: $R \subseteq X \times X$ is a branching bisimulation relation if the following rules hold for R.

$$\frac{q \to_a q' \qquad R(q,p)}{\exists p', p''(p \twoheadrightarrow_\tau p' \to_a p'' \wedge R(q,p') \wedge R(q',p''))}\,\mathrm{bis}_b$$

$$\frac{q \to_\tau q' \qquad R(q,p)}{R(q',p) \vee \exists p', p''(p \twoheadrightarrow_\tau p' \to_\tau p'' \wedge R(q,p') \wedge R(q',p''))}\,\mathrm{bis}_{b\tau}$$

$$\frac{R(p,q)}{R(q,p)}\,\mathrm{symm}$$

States q, p are branching bisimilar, $q \underline{\leftrightarrow}_b\, p$ if there exists a branching bisimulation relation R such that $R(q,p)$.

Branching apartness

We define branching apartness by transporting the rules for branching bisimulation to rules for $Q \subseteq X \times X$ where $\neg Q$ is a branching bisimulation.

Definition. $Q \subseteq X \times X$ is a branching apartness in case the following rules hold for Q.

$$\frac{q \to_a q' \qquad \forall p', p''(p \twoheadrightarrow_\tau p' \to_a p'' \implies Q(q,p') \vee Q(q',p''))}{Q(q,p)}\,\mathrm{in}_b$$

$$\frac{q \to_\tau q' \quad Q(q',p) \quad \forall p', p''(p \twoheadrightarrow_\tau p' \to_\tau p'' \implies Q(q,p') \vee Q(q',p''))}{Q(q,p)}\,\mathrm{in}_{b\tau}$$

$$\frac{Q(p,q)}{Q(q,p)}\,\mathrm{symm}$$

States q and p are branching apart, $q \mathrel{\underline{\#}_b} p$, if for all branching apartness relations Q, we have $Q(q,p)$.

Branching bisimulation and branching apartness

- By definition: Q is a branching apartness iff $\neg Q$ is a branching bisimulation, so
- $q \mathrel{\underline{\#}_b} p$ if and only if $\neg(q \mathrel{\underline{\leftrightarrow}_b} p)$.
- As $q \mathrel{\underline{\#}_b} p$ is an inductive notion, we have that $q \mathrel{\underline{\#}_b} p$ if and only it is derivable using the derivation rules (symm) and the following two:

$$\frac{q \to_a q' \qquad \forall p', p''(p \twoheadrightarrow_\tau p' \to_a p'' \implies q \mathrel{\underline{\#}_b} p' \vee q' \mathrel{\underline{\#}_b} p'')}{q \mathrel{\underline{\#}_b} p}\,\mathrm{in}_b$$

$$\frac{q \to_\tau q' \quad q' \mathrel{\underline{\#}_b} p \quad \forall p', p''(p \twoheadrightarrow_\tau p' \to_\tau p'' \implies q \mathrel{\underline{\#}_b} p' \vee q' \mathrel{\underline{\#}_b} p'')}{q \mathrel{\underline{\#}_b} p}\,\mathrm{in}_{b\tau}$$

Example

$$\begin{tikzcd}
 & s \arrow[dl, "\tau"'] \arrow[d, "c"] \arrow[dr, "d"] & \\
s_1 \arrow[d, "c"'] & s_2 & s_3 \\
s_4 & &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & r \arrow[dl, "\tau"'] \arrow[d, "d"] \\
r_1 \arrow[d, "c"'] & r_2 \\
r_3 &
\end{tikzcd}$$

We give a derivation of $s \ \underline{\#}_b \ r$:

$$\dfrac{s \rightarrow_c s_2 \qquad [r \rightarrow_\tau r_1 \rightarrow_c r_3] \qquad \dfrac{\dfrac{s \rightarrow_d s_3 \qquad \checkmark}{s \ \underline{\#}_b \ r_1}}{s \ \underline{\#}_b \ r_1 \vee s_2 \ \underline{\#}_b \ r_3}}{s \ \underline{\#}_b \ r}$$

NB: Remember the derivation rule:

$$\dfrac{q \rightarrow_a q' \qquad \forall p', p''(p \twoheadrightarrow_\tau p' \rightarrow_a p'' \implies q \ \underline{\#}_b \ p' \vee q' \ \underline{\#}_b \ p'')}{q \ \underline{\#}_b \ p} \ \mathrm{in}_b$$

Proving that $\underline{\leftrightarrow}_b$ is an equivalence relation

This is remarkably tricky, because if R_1, R_2 are branching bisimulation relations, then $R_1 \circ R_2$ need not be a branching bisimulation relation. (So the "obvious" proof of transitivity fails.)

Basten used the notion of semi-branching bisimulation relation and proved that (1) "being semi-branching bisimilar", $\underline{\leftrightarrow}_{sb}$, is an equivalence relation and (2) $\underline{\leftrightarrow}_{sb}$ coincides with $\underline{\leftrightarrow}_b$.

We similarly introduce semi-branching apartness relation, $\underline{\#}_{sb}$, by replacing rule $(\mathrm{in}_{b\tau})$ by

$$\frac{\begin{array}{c} q \rightarrow_\tau q' \qquad q' \,\underline{\#}_{sb}\, p \\ \forall p', p''(p \twoheadrightarrow_\tau p' \rightarrow_\tau p'' \implies q' \,\underline{\#}_{sb}\, p'' \vee (q \,\underline{\#}_{sb}\, p' \wedge q \,\underline{\#}_{sb}\, p'')) \end{array}}{q \,\underline{\#}_{sb}\, p}\ \mathrm{in}_{sb\tau}$$

So, $q \,\underline{\#}_{sb}\, p$ in case this is derivable by these adapted set of rules.

Proving the co-transitivity of branching apartness

The proof of co-transitivity of $\underline{\#}_b$ (and thus that $\underline{\leftrightarrow}_b$ is an equivalence relation) proceeds in the following steps.

1. We prove $q \underline{\#}_{sb} p \implies q \underline{\#}_b p$ (by induction on $q \underline{\#}_{sb} p$).
2. We prove a number of basic lemmas for $\underline{\#}_{sb}$.
   (Typically useful results we would also like to have for $\underline{\#}_b$, but we can't obtain directly for $\underline{\#}_b$.)
3. We prove the apartness stuttering property for $\underline{\#}_{sb}$.
4. We prove that $q \underline{\#}_b p \implies q \underline{\#}_{sb} p$ (by induction on $q \underline{\#}_{sb} p$, using the apartness stuttering property) and we conclude that $\underline{\#}_b = \underline{\#}_{sb}$.
5. We prove co-transitivity for $\underline{\#}_b$ using the lemmas under (2).

For one of the basic lemmas under (2) we move over to the "bisimulation view", as the result seems easier to obtain there.

Stuttering and apartness stuttering

The stuttering property states that the following holds (for $\underline{\leftrightarrow}_b$)

$$\frac{r \to_\tau r_1 \to_\tau \ldots \to_\tau r_n \to t \quad (n \geq 0) \qquad r \underline{\leftrightarrow}_b p \qquad t \underline{\leftrightarrow}_b p}{\forall i\,(1 \leq i \leq n)\; r_i \underline{\leftrightarrow}_b p}$$

If we cast this as a property about apartness we obtain the following apartness stuttering property

$$\frac{r \twoheadrightarrow_\tau q \twoheadrightarrow_\tau t \qquad q \mathrel{\underline{\#}} p}{r \mathrel{\underline{\#}} p \vee t \mathrel{\underline{\#}} p}\,\mathrm{stut}$$

LEMMA. The apartness stuttering property holds for $\underline{\#}_{sb}$
PROOF. By induction on $q \mathrel{\underline{\#}_{sb}} p$ (using various auxiliary properties).

Variations on the rules

We can show that other rules are sound for proving apartness, for example (thanks to David Jansen):

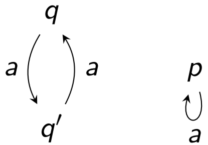
$$\frac{q \rightarrow_a q' \qquad \forall p', p''(p \twoheadrightarrow_\tau p' \rightarrow_a p'' \implies \color{red}{p \mathrel{\underline{\#}_b} p'} \lor q' \mathrel{\underline{\#}_b} p'')}{q \mathrel{\underline{\#}_b} p}\ \mathrm{in}_b^A$$

Or, combining bisimulation and apartness, the following rule is sound:

$$\frac{q \rightarrow_a q' \qquad \forall p', p''(p \twoheadrightarrow_\tau p' \rightarrow_a p'' \land q' \mathrel{\underline{\leftrightarrow}_b} p'' \implies q \mathrel{\underline{\#}_b} p')}{q \mathrel{\underline{\#}_b} p}\ \mathrm{in}_b^{\underline{\leftrightarrow}_b}$$

Using $\underline{\#}_b$ to prove $q \underline{\leftrightarrow}_b p$

Example.



We search for the shortest derivation of $q \underline{\#}_b p$ and notice it doesn't exist, and therefore we can conclude that $\neg q \underline{\#}_b p$ and so $q \underline{\leftrightarrow}_b p$. In our search for a deduction we keep track of goals that we have already encountered.

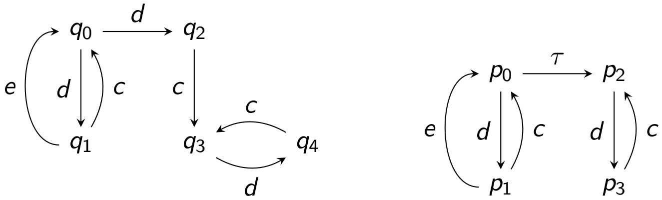
$$\dfrac{q \rightarrow_a q' \qquad \dfrac{\dfrac{q' \rightarrow_a q \qquad \dfrac{\text{fail}}{q' \underline{\#}_b p \vee q \underline{\#}_b p}}{q' \underline{\#}_b p}}{q \underline{\#}_b p \vee q' \underline{\#}_b p}}{q \underline{\#}_b p}$$

From $q \underline{\#}_b p$ to a distinguishing formula (example)

$$
\begin{tikzcd}
q_0 \arrow[r, "d"] \arrow[d, "d"'] & q_2 \arrow[d, "c"'] & \\
q_1 \arrow[u, bend right, "c"'] \arrow[u, bend left=60, "e"] & q_3 \arrow[r, bend right, "d"'] & q_4 \arrow[l, bend right, "c"']
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
p_0 \arrow[r, "\tau"] \arrow[d, "d"'] & p_2 \arrow[d, "d"'] \\
p_1 \arrow[u, bend right, "c"'] \arrow[u, bend left=60, "e"] & p_3 \arrow[u, bend right, "c"']
\end{tikzcd}
$$

$$
\dfrac{q_0 \to_d q_2 \quad \dfrac{\dfrac{\dfrac{\dfrac{p_1 \to_e p_0}{p_1 \underline{\#}_b q_2}}{q_2 \underline{\#}_b p_1}}{q_0 \underline{\#}_b p_0 \lor q_2 \underline{\#}_b p_1} \quad \dfrac{\dfrac{q_0 \to_d q_1 \quad \dfrac{\dfrac{\dfrac{q_1 \to_e q_0}{q_1 \underline{\#}_b p_3}}{q_0 \underline{\#}_b p_2 \lor q_1 \underline{\#}_b p_3}}{\forall p', p''(p_2 \twoheadrightarrow_\tau p' \to_d p'' \implies q_0 \underline{\#}_b p' \lor q_1 \underline{\#}_b p'')}}{q_0 \underline{\#}_b p_2}}{q_0 \underline{\#}_b p_2 \lor q_2 \underline{\#}_b p_3}}{\forall p', p''(p_0 \twoheadrightarrow_\tau p' \to_d p'' \implies q_0 \underline{\#}_b p' \lor q_2 \underline{\#}_b p'')}}{q_0 \underline{\#}_b p_0}
$$

From $q \mathrel{\underline{\#}_b} p$ to a distinguishing formula (example)



- Korver has given an algorithm that generates an HMLU (Hennessy-Milner Logic with Until) formula Φ that distinguishes two states s and t in case $\neg(s \mathrel{\underline{\leftrightarrow}_b} t)$.
- We can extract such a formula from a derivation of $s \mathrel{\underline{\#}_b} t$.

For the example, the formula derived from the derivation of $q_0 \mathrel{\underline{\#}_b} p_0$ is

$$\Phi := (\mathrm{tt}\,\langle d\rangle\,(\mathrm{tt}\,\langle e\rangle\,\mathrm{tt}))\ \langle d\rangle\ \neg(\mathrm{tt}\,\langle e\rangle\,\mathrm{tt})$$

We have $q_0 \models \Phi$ and $p_0 \not\models \Phi$.

Questions?