Bisimulation: entre Théorie des Modèles de la Logique Modale et Vérification

Raphaëlle Crubillé
mail: raphaelle.crubille@lis-lab.fr

AMU-LIS

Master IMD

Théorie des modèles.

Origine (comme champ de recherche propre): Löwenheim-Skolem '15, Tarski'54.

Idée générale:

Étudier la relation entre:

- un langage logique (logique du premier ordre, logique modale,...);
- les modèles pour cette logique. (sémantique de Tarski, structures de Kripke,...).

Lignes directrices:

- quelles **propriétés structurelles** sont exprimables dans la logique ?
- quelles **classes de structures** sont définissables dans la logique ?

Exemple d'outils considérés:

- construction de modèles;
- analyse de modèles pour une formule logique/ théorie donnée
- équivalence de structures par rapport à la notion de satisfiabilité
- phénomènes de préservation...

Rappel: logique du premier ordre.

Signature Σ:

- Symboles de fonctions (avec arité n)
- Symboles de prédicats (relations entre éléments).

e.g. $(1, \times)$; $(1, 0, +, \times, <)$...

Formules

$\phi, \psi := \top \mid \bot \mid \phi \to \psi \mid \forall x.\psi \mid \exists x.\psi \mid \phi \wedge \psi \mid P(t_1, \ldots, t_n) \mid t_1 = t_2.$

Modèle d'une théorie logique du premier ordre

On fixe Σ: une signature. Un modèle:

- X un ensemble,
- pour tout f, symbole de function d'arité n: $\widetilde{f} : X^n \to X$;
- pour tout P, symbole de prédicat d'arité n: $\widetilde{P} \subseteq X^n$.

Example (Propriété structurelle exprimable)

Comment exprimer **dans la logique** le fait d'être un groupe ?
$\Sigma = (\times, 1)$. On cherche ϕ telle que $G \models \phi$ ssi G est un groupe.

$$\phi := (\forall x, x \times 1 = x) \wedge (\forall x \exists y x \times y = 1).$$

$\Rightarrow$ la propriété "être un groupe" est définissable.

Example

Signature: $(1, 0, +\times, <)$. $(\mathbb{R} +$ opérations usuelles$)$: modèle pour cette signature.

Question: Quelles sont les structures **logiquement** équivalentes à $\mathbb{R}$?

Théorie des modèles pour la logique du premier ordre (2)

Theorem (Tarski'30)

*Les structures **logiquement** équivalentes à $\mathbb{R}$ sont les **corps réels clos** (e.g. réels calculables, réels algébriques...)*
Axiomes:

- *axiomes des corps ordonnés*
- *chaque nombre positif a une racine carré*
- *pour chaque $d \in \mathbb{N}$ impair, un polynome de degré d a au moins une racine.*

Application de la théorie des modèles aux mathématiques générales

- Dans les années '40 (Tarski, Mal'tsev, Robinson...): Des théorèmes de théorie des groupes ont été obtenus par des méthodes de théorie des modèles.
- ...

Théorie des modèles de la logique modale.

Questions

- Les théorèmes "standards" de la théorie des modèles de FOL peuvent-ils être généralisés ?
- Quel est le **pouvoir expressif** de la logique modale ?
- Qu'est ce qui, d'un point de vue **sémantique** rend une logique **modale** ?

Spécificités de la logique modale:

- La logique modale peut être encodée en logique du premier ordre (voir + tard).

  $\Rightarrow$ un certain nombre de résultats de la théorie des modèles de FOL peuvent être étendus à la logique modale.
- Beaucoup des problématiques de théorie des modèles peuvent être abordés avec la notion de **bisimulation**.

Système réactifs

Systèmes qui intéragissent en continu avec l'environnement:

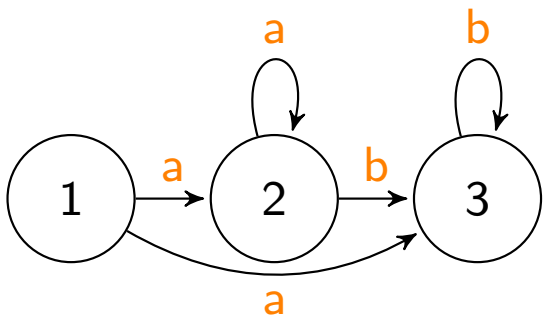
- l'environement (quand il le décide) envoie des signaux au système
- le système réagit à ses signaux

Example

- une machine à café
- un système de contrôle automatique dans un avion...

Un système réactif peut être modélisé par un **système de transition labellé (LTS)**

$\mathcal{M} :=$



Objectif:

Étant donné un système, on veut **prouver** qu'il vérifie une **spécification.**

Cas particulier:

Cette spécification peut s'exprimer comme une **équivalence** de systèmes.
Par exemple:

- la spécification s'exprime comme un LTS **idéal** (potentiellement non-déterministe), et l'implémentation s'exprime comme un LTS (déterministe) issu d'un language de programmation concret.
- On veut prouver la **correction** d'optimisations (par exemple effectuéee par un compilateur), c'est à dire que le système optimisé a le même comportement que le système non optimisé.

Équivalences entre systèmes réactifs

Définition informelle:

Deux systèmes $\mathcal{L}_1$, $\mathcal{L}_2$ sont **équivalents** si aucun **environnement extérieur** n'est capable de les distinguer l'un de l'autre.

Problème:

Cette définition dépend de comment l'environnement est **modélisé**.

- Il faut fixer un modèle d'environement (ou "adversaire") (qui fixe le **pouvoir expressif** de l'adversaire)
  $\Rightarrow$ on obtient une notion **d'équivalence contextuelle:** $\sim^{ctx}$.
- Mais on garde une **quantification universelle** sur tous les environements, qui rend la définition difficile à uttiliser en pratique.

Definitions intrinsèques

- Équivalence de trace: $\sim^{tr}$
- Bisimilarité: $\sim^{bs}$

La bisimilarité est **plus exigeante** que l'équivalence de trace:
$\mathcal{L}_1 \sim^{bs} \mathcal{L}_2 \quad \Rightarrow \quad \mathcal{L}_1 \sim^{tr} \mathcal{L}_2$.

Correction d'un modèle d'adversaire par rapport à une équivalence intrinsèque:

En fonction du modèle d'adversaire, on peut avoir:

$$\mathcal{L}_1 \sim^{tr} \mathcal{L}_2 \quad \Rightarrow \quad \mathcal{L}_1 \sim^{ctx} \mathcal{L}_2$$
$$\text{ou } \mathcal{L}_1 \sim^{bs} \mathcal{L}_2 \quad \Rightarrow \quad \mathcal{L}_1 \sim^{ctx} \mathcal{L}_2.$$

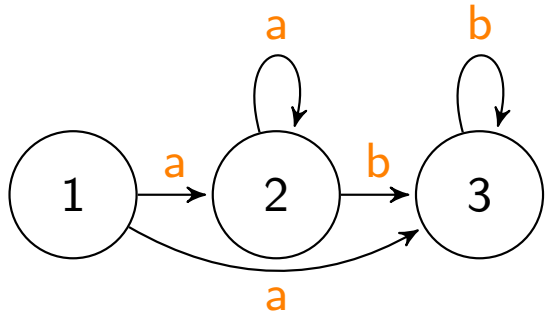
Example (Programmes concurrents)

CCS: language idéal pour la programmation concurrente.

- Les communications entre processus se font par synchronisation des processus sur des canaux.
- On peut modéliser la sémantique par un système de transition labellée.

En fonction du modèle d'adversaire choisi, on peut montrer correcte l'équivalence de trace ou la bisimilarité.

Récapitulatif du cours précédent (1)

$\mathcal{M}:=$ 

$+$ *val* : Etats x Variables $\rightarrow \{\text{true}, \text{false}\}$.

Modèle de Kripke pour LM

Équivalence Modale:
$(\mathcal{M}, w) \equiv^{LM} (\mathcal{M}', w')$ quand $\forall \phi$ LM-formula, $\mathcal{M}, w \models \phi \Leftrightarrow \mathcal{M}', w' \models \phi$.

Système de Transition Labellés (LTS)

modélisent des systèmes réactifs

Bisimulation: équivalence structurelle sur les LTSs.

Objectifs de ce cours:

- Peut-on comparer l'équivalence modale et la bisimulation ?
- Uttilisation de bisimulations pour des problèmes de théorie des modèles: construction de modèles, résultats de (non)-définissabilité

Récapitulatif du cours précédent (2)

Theorem (Invariance par bisimulation)

Soit (M, w_1) (N, w_2) deux structures de Kripke pointées. Si il existe une bisimulation R entre (M, w_1) et (M, w_2), alors $Th(M, w_1) = Th(M, w_2)$.

Consequences:

- Construction de modèles:
  e.g. toute formule ϕ satisfiable est satisfiable **à la racine d'un arbre**.
- (Non)-définissabilité: toute classe de structures définissable doit être **stable par bisimulation**.

Aujourd'hui

- Théorème d'Hennessy-Milner: une réciproque partielle du théorème d'invariance
- Théorème (Construction de modèle):
  toute formule ϕ satisfiable est satisfiable à la racine d'un arbre **fini**.

Préliminaires: un peu de théorie des points fixes.

Ensembles partiellement ordonnés.

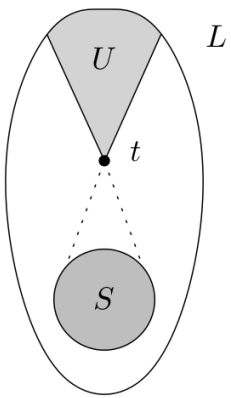
Definition

- Un **ensemble préordonné** (poset) L est un ensemble non vide équippé avec une relation sur ses éléments qui est un ordre partiel $\leq$ (i.e. réflexif, transitif, anti-symmétrique).
- Le **supremum** d'une partie $S \subseteq L$ est le plus petit element $x \in L$, tel que pour tout $y \in S$, $y \leq x$.
- L' **infinum** d'une partie $S \subseteq L$ est le plus grand element $x \in L$, tel que pour tout $y \in S$, $x \leq y$.

Example

$\mathbb{N}, \leq$, où $n \leq m$ si n divise m.

Example (Infinum et supremum dans un poset)

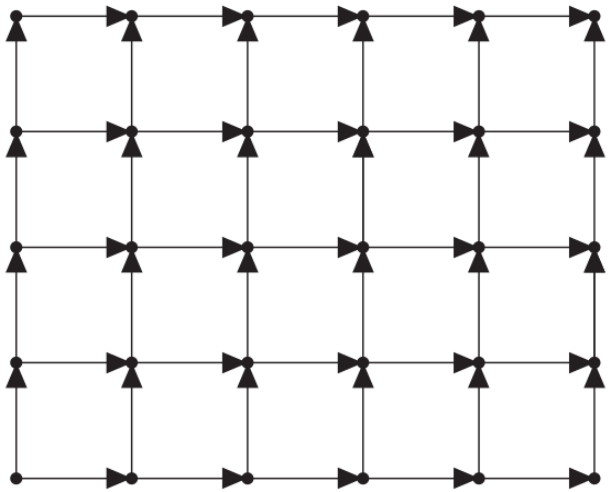


Treillis complet.

Definition

Un **treillis complet** est un poset tel que tout $S \subseteq L$ a un supremum.

Example (Un treillis complet/ complete lattice)



Example

L'intervalle $[0, 1]$ (avec l'ordre usuel sur $\mathbb{R}$).

Théorème du point fixe de Tarski

Definition

L un poset, $f : L \to L$.

- f est croissante si pour tout $x \leq y \in L$, $f(x) \leq f(y)$.
- $x \in L$ est un pre-fixpoint de f si $f(x) \leq x$, et un post-fixpoint si $x \leq f(x)$.
- $x \in L$ est un point fixe de L quand $f(x) = x$.

Theorem (Théorème du point fixe)

Sur un treillis complet L, une fonction croissante $F : L \to L$ a un treillis complet de point fixes. En particulier, le plus petit point fixe de la fonction $lfp(f)$ est l'infinum de ses pre-fixpoints, et le plus grand point fixe $gfp(f)$ est le supremum de ses post-fixpoints.

Example

Au tableau

Caractérisation de la Bisimilarité comme un point fixe.

La bisimilarité vu comme un point fixe (2).

On fixe une SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$.
$\mathcal{R} := \{S \mid S \subseteq W \times W \text{ s.t. } wSw' \Rightarrow val(s) = val(s')\}$.

Definition (Un opérateur sur les relations binaires)

On définit $F : \mathcal{R} \to \mathcal{R}$ par:

$$F(S) = \{(s,t) \mid \forall \alpha \in \tau, \forall s' \in W, s \xrightarrow{\alpha} s' \Rightarrow \exists t', t \xrightarrow{\alpha} t' \wedge s' S t'$$
$$\text{et } \forall t' \in W, t \xrightarrow{\alpha} t' \Rightarrow \exists s', s \xrightarrow{\alpha} s' \wedge s' S t'$$
$$\text{et } \forall \phi \in \Psi, \phi(t) = \phi(s)\}.$$

Remarque

S est une bisimulation si et seulement si $S \subseteq F(S)$, i.e. un **post-fixpoint** de F.

Lemma

F est monotone, sur le lattice complet $\mathcal{R}$.

Theorem

- $\sim$ *est le plus* ***grand*** *point fixe de* F.
- $\sim$ *est le plus grand élément* S *de* $\mathcal{R}$ *tel que* $S \subseteq F(S)$.

Continuité et co-continuité

Definition

L un treillis complet, $f : L \to L$.

- f est **continu** si pour toute suite croissante $x_1 \leq x_2 \leq \ldots$ d'éléments de L, $\sup_{i \in \mathbb{N}}(f(x_i)) = f(\sup_{i \in \mathbb{N}} x_i)$;
- f est **co-continu** si pour toute suite décroissante $x_1 \geq x_2 \geq \ldots$ d'éléments de L, $\inf_{i \in \mathbb{N}}(f(x_i)) = f(\inf_{i \in \mathbb{N}} x_i)$;

Notation: Itérés d'une fonction

L un treillis complet, $f : L \to L$, $x \in L$.

$$f^0(x) := x;$$
$$f^{n+1}(x) := f(f^n(x))$$

Theorem (Théorème de Continuity - Co-continuity)

- *si f est continu*, $lfp(f) = \sup\{f^n(\bot) \mid n \in \mathbb{N}\}$;
- *si f est co-continu*, $gfp(f) = \inf\{f^n(\top) \mid n \in \mathbb{N}\}$

On fixe une SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$.

Definition (Stratification de la Bisimilarité (on $\mathbb{N}$))

Par induction sur $n \in \mathbb{N}$:

$$\begin{aligned} \sim_0 &:= \{(s,t) \in W \times W \mid val(s) = val(t)\} = F(\top) \\ \sim_{n+1} &:= F^n(\top) = F(\sim_n) \\ \sim_\omega &:= \inf_{n\in\mathbb{N}} F^n(\top) = \cap_{n\in\mathbb{N}} \sim_n . \end{aligned}$$

Remarque:

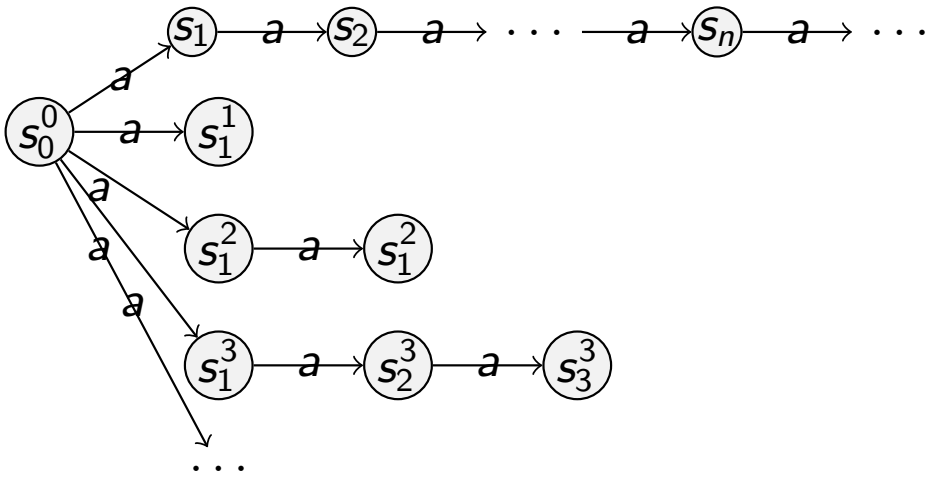
- $F(\sim_\omega) \subseteq \sim_\omega$, i.e. $\sim_\omega$ est un pré-fixpoint pour F.
- $\sim \subseteq \sim_\omega$ (parce que $\sim_\omega \leq \top$, donc $\sim_\omega = F(\sim_\omega) \leq F(\top)$, et en itérant on a $\forall n \in \mathbb{N}, \sim_\omega \leq F^n(\top)$, et donc $\sim_\omega \leq \inf_{n\in\mathbb{N}} F^n(\top)$)
  mais en général, $\sim_\omega \neq \sim$

Theorem

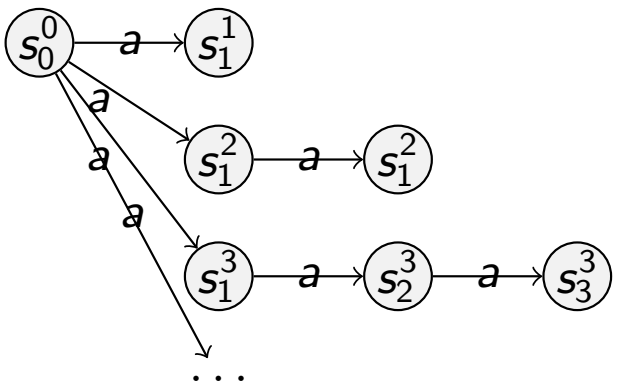
Si la SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$ est ***à branchements finis****, alors F est co-continu, et $\sim = \sim_\omega$*

Comparer la bisimilarité et son ω-approximation.

$\mathcal{L}_1$:



$\mathcal{L}_2$:



- $(\mathcal{L}_1, s_0) \sim^n (\mathcal{L}_2, s_0^0)$.
- $(\mathcal{L}_1, s_0) \not\sim (\mathcal{L}_2, s_0^0)$.

Definition

$s \equiv_n^{LM} t$ quand pour tout ϕ de profondeur modale **au plus n**, $s \models \phi \Leftrightarrow t \models \phi$.

On voit immédiatement que $(\equiv^{LM}) = (\cap_{n\in\mathbb{N}} \equiv_n^{LM})$.

Theorème (Hennessy-Milner)

Soit une SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$. On suppose Ψ, τ finite. Alors $\equiv_n^{LM} = \sim_n$, et $\equiv^{LM} = \sim_\omega$.

Proof.

Formule charactéristique: $s \in W$:

$$\phi_0(s) = \bigwedge_{p\in\Psi | val(s)(p)=\top} p \wedge \bigwedge_{p\in\Psi | val(s)(p)=\bot} \neg p.$$

$$\phi_{n+1}(s) = \phi_0(s) \wedge \bigwedge_{(s',\alpha) | s \xrightarrow{\alpha} s'} \Diamond_\alpha \cdot \phi_n(s') \wedge \bigwedge_{\alpha\in\tau} \Box_\alpha \cdot \vee_{s' | s \xrightarrow{\alpha} s'} \phi_n(s').$$

Pour voir que les $\bigwedge$ et $\bigvee$ sont finis, on uttilise qu'il y a un nombre fini de formule de LM de profondeur modale $\leq n$ □

$$\phi_0(s) = \bigwedge_{p\in\Psi | val(s)(p)=\top} p \wedge \bigwedge_{p\in\Psi | val(s)(p)=\bot} \neg p.$$

$$\phi_{n+1}(s) = \phi_0(s) \wedge \bigwedge_{(s',\alpha)|s\xrightarrow{\alpha}s'} \Diamond_\alpha \cdot \phi_n(s') \wedge \bigwedge_{\alpha\in\tau} \Box_\alpha \cdot \vee_{s'|s\xrightarrow{\alpha}s'} \phi_n(s').$$

On prouve par induction sur n:

(H_n): $s \sim^n t \Leftarrow t \models \phi_n(s)$

- $n = 0$: on doit prouver $t \models \phi_0(s) \Rightarrow \forall p \in \Psi, p(s) = p(t)$
- On suppose (H_n), et $t \models \phi_{n+1}(s)$. On doit prouver $(s,t) \in F(\sim^n)$:
  - $\forall p \in \Psi, p(s) = p(t)$: c'est parce que $t \models \phi_0(s)$;
  - Soit s', $\alpha \in \tau$ tel que $s \xrightarrow{\alpha} s'$. Alors $(t \models \phi_{n+1}(s)) \Rightarrow (t \models \Diamond_\alpha \phi_n(s')) \Rightarrow (\exists t' \text{ t.q. } t' \models \phi_n(s')) \Rightarrow (\exists t' \text{ t.q. } (t', s') \in \sim^n)$.
  - Soit $t', \alpha \in \tau$ tel que $t \xrightarrow{\alpha} t'$. Alors $(t \models \phi_{n+1}(s)) \Rightarrow (t \models \Box_\alpha \vee_{s'|s\xrightarrow{\alpha}s'} \phi_n(s')) \Rightarrow (\exists s' \text{ t.q. } t' \models \phi_n(s')) \Rightarrow (\exists s' \text{ t.q. } (t', s') \in \sim^n)$.

Definition

$s \equiv_n^{LM} t$ quand pour tout ϕ de profondeur modale **au plus n**, $s \models \phi \Leftrightarrow t \models \phi$.

On voit immédiatement que $(\equiv^{LM}) = (\cap_{n\in\mathbb{N}} \equiv_n^{LM})$.

Theorème (Hennessy-Milner)

Soit une SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$. On suppose Ψ, τ finite. Alors $\equiv_n^{LM} = \sim_n$, et $\equiv^{LM} = \sim_\omega$.

Corollaire

Si la SK $(W, (R_\alpha)_{\alpha\in\tau}, val)$ est **à branchements finis**, alors l'équivalence de la logique modale et la bisimilarité coincide.

Theorem (Modèle arbre borné)

Si Ψ, τ sont finis. Toute formule ϕ satisfiable est satisfiable à la racine d'un ***arbre fini****, de taille $\leq f(\mathit{length}(\phi))$.*

Proof.

Toute structure de Kripke pointé est n-bisimilar à un arbre fini telle que à chaque neud, l'ensemble de ses fils est de cardinal $\leq 2^{\#\{\psi | \mathsf{length}(\psi) \leq n\}}$. Pour prouver cela, on construit une structure de Kripke par:

1. on fait l'unfolding de la structure de Kripke à partir de l'état initial (on obtient un arbre potentiellement de profondeur ∞, et à branchements ∞).
2. on coupe la profondeur de l'arbre à n;
3. à chaque étage, on collapse les états qui sont dans la même classe d'équivalence pour $\sim_n$. On peut borner le nombre de classe d'équivalence en uttilisant que $(s \equiv_n^{LM} t) \Leftrightarrow (s \sim_n t)$.

□

Application : Finite tree model property

Theorem (Modèle arbre borné)

Si Ψ, τ sont finis. Toute formule ϕ satisfiable est satisfiable à la racine d'un ***arbre fini****, de taille $\leq f(length(\phi))$.*

Corollary

Si Ψ, τ sont finis, la satisfiabilité d'une formule de $LM(\tau, \phi)$ est un problème décidable (cf bounded model property).

Remarque

En logique du premier ordre:

- ce n'est pas le cas que toute formule satisfiable est satisfiable dans un modèle fini;
- La satisfiabilité n'est pas décidable

Mais ces deux propriétés sont vrais dans FO^2 (logique du premier ordre avec seulement 2 variables).