



HAL
open science

Modélisation de prédicats dans la syntaxe propositionnelle

Arnaud Kohler

► **To cite this version:**

| Arnaud Kohler. Modélisation de prédicats dans la syntaxe propositionnelle. 2021. ⟨hal-03195162v2⟩

HAL Id: hal-03195162

<https://hal.science/hal-03195162v2>

Preprint submitted on 21 May 2021 (v2), last revised 29 Oct 2022 (v7)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Modélisation de prédicats dans la syntaxe propositionnelle

Arnaud Kohler

Arnaud.kohler@pacariane.com

Résumé

Il existe un nombre conséquent de langages formels pour modéliser le raisonnement. Chacun capture plusieurs propriétés, mais aucun ne réussit à les couvrir toutes. Nous avons proposé une solution alternative avec la logique contextuelle. Sa particularité est de se maintenir strictement dans le cadre syntaxique monotone de la logique propositionnelle tout en bénéficiant d'une fonction d'interprétation sémantique non monotone. Elle permet la modélisation et l'exploitation de connaissances incohérentes ou modales épistémiques. Nous présentons dans cet article comment utiliser ses concepts pour capturer la notion de prédicat.

Abstract

There are a number of formal languages for modelling reasoning. Each captures several properties, but none succeeds in covering them all. We have proposed an alternative solution with contextual logic. Its particularity is to keep strictly within the monotonic syntactic framework of propositional logic while benefiting from a non-monotonic semantic interpretation function. It allows the modelling and exploitation of incoherent or modal epistemic knowledge. In this paper we present how to use its concepts to capture the notion of predicate.

Introduction

L'Intelligence Artificielle symbolique est confrontée à une difficulté majeure : bénéficier d'un formalisme unique pour modéliser les différents modes de raisonnement humain (cf. par exemple Daniel Andler, 2004). Les recherches sur ce sujet ont conduit à une multiplication des langages formels (logiques modales, para-cohérentes, des prédicats, multivaluées, adaptatives, etc.). Chaque formalisme capture une propriété particulière, sans toutefois modéliser l'ensemble des modes de raisonnement constatés empiriquement chez l'humain.

Dans ce cadre, nous avons présenté (cf. Arnaud Kohler, 2019) la logique contextuelle L_c . Tout en respectant strictement la syntaxe monotone de la logique propositionnelle L_p , elle bénéficie d'une fonction d'interprétation sémantique qui supporte la modélisation de connaissances incohérentes ou modales épistémiques. Nous rappelons ses principes dans un premier temps. Nous présentons ensuite comment l'utiliser pour capturer la notion de prédicat.

La logique contextuelle

L_c est construite sur le postulat suivant, dit postulat contextuel (cf. Arnaud Kohler, 2019) :

Soit L un langage formel muni des fonctions de production syntaxique \vdash_L et d'interprétation syntaxique \models_L . Une formule bien formée f de L est un ensemble de signes qui n'a pas de signification. Son sens est porté par une pensée, qui est une proposition élémentaire de L « *qui ne se prononce pas* ». Pour c symbolisant cette pensée, la relation entre c et f est $c \models_L f$.

Nous appelons logique contextuelle L_c l'application de ce postulat à la logique propositionnelle L_p . L'expression $c \models_{L_p} f$, équivalente à $\models_{L_p} c \rightarrow f$, n'affirme ni la pensée c ni la phrase f , mais la connaissance que la phrase f exprime la pensée c . Les propositions c respectent les propriétés syntaxiques de L_p . Les expressions non contextuelles sont acceptées, mais elles sont considérées comme un ensemble de signes sans signification, permettant la production \vdash_{L_p} mais pas l'interprétation sémantique \models_{L_p} . Techniquement, cela revient à les rendre inutiles.

Toutes les formules utiles prennent donc une forme $c_i \rightarrow f_i$. Les expressions dans L_c sont naturellement ordonnées. Au rang 0, on retrouve les formules constituées de propositions de L_p , au rang 1 les pensées qui sont exprimées par des formules de rang 0, et au rang n des pensées qui portent des formules de rang au plus $n-1$.

L_c a été présentée dans Arnaud Kohler, 2019, ainsi que les éléments de comparaisons avec les autres langages formels. Classiquement, les formalismes s'attachent à modéliser des connaissances réputées « vraies ». Afin de gérer les incohérences et les incertitudes sémantiques, ils sont enrichis pour exprimer des niveaux de nécessité (les logiques modales épistémiques), de quantification (les logiques des prédicats), d'incohérence (les logiques para-

consistantes), de nouvelles règles de production syntaxique (la logique des défauts par exemple) ou de stratégies multiples d'interprétation (les logiques adaptatives).

En logique contextuelle, l'ensemble composé des négations des pensées caractérise un modèle possible de tout ensemble de connaissances. Dit autrement, dans L_c , « toute pensée est possiblement fausse, et toute proposition de rang 0 peut être vraie ou fausse » : l'incohérence et l'incertitude sont inscrites intrinsèquement dans la syntaxe. La première conséquence est qu'une théorie contextuelle est toujours consistante (elle admet au moins un modèle).

La seconde conséquence est que la logique contextuelle est dans l'incapacité d'exprimer une certitude. Pour remédier à ce problème, L_c propose de relativiser l'interprétation sémantique à un ensemble de pensées, appelé un contexte : on ne dit pas « f est vraie (ou fausse) », mais « f est vraie (ou fausse) dans ce contexte ». Une fonction identifie alors les contextes considérés comme pertinents pour l'interprétation sémantique. Soient E_{L_c} une théorie de L_c , et i et j deux entiers tels que $0 < i < j$. Les contextes pertinents s'obtiennent de la manière suivante :

- calcul des contextes, dits maximaux, de rang j et plus – c'est-à-dire les conjonctions maximales de pensées de rang j et plus vérifiant au moins un modèle de E_{L_c} ,
- puis enrichissement de chaque contexte maximal C_m , par le contexte, dit crédible, qui lui est associé sur le rang i à $j-1$ – c'est-à-dire la conjonction des pensées de rang i à $j-1$ vraies dans tous les modèles de $\{E_{L_c}, C_m\}$.

Nous obtenons ainsi un ensemble de contextes, que nous appelons les contextes épistémiques. Ils identifient des perspectives sémantiques : l'interprétation est réalisée en considérant les productions de chacun des contextes épistémiques sur les rangs 0 à $i-1$. Une expression f est réputée vraie si elle est produite par au moins un contexte épistémique. Notons qu'une perspective est toujours consistante unitairement, mais que f et $\neg f$ peuvent être vraies selon deux perspectives différentes.

i et j peuvent prendre théoriquement n'importe quelles valeurs. Nous utilisons dans la suite de ce document les seuils 2 et 3, suffisants pour couvrir le niveau d'expressivité attendu.

L_c respecte la syntaxe de la logique propositionnelle, et est donc syntaxiquement monotone. Elle est sémantiquement non monotone, adoptant une conception faillibiliste (une expression n'est vraie que tant qu'elle n'est pas contredite) et perspectiviste (l'interprétation sémantique s'obtient en considérant les interprétations, éventuellement contradictoires, de chaque contexte épistémique). Pour conclure cette présentation, notons que L_c apporte une capacité réflexive au raisonnement en modélisant une relation entre la pensée c et la phrase f qui l'exprime.

Utilisation de L_c pour modéliser des connaissances modales épistémiques

Supposons dans L_p un ensemble de propositions $\{a, b, c, d\}$, et la théorie suivante :

$$E_{L_p} = \{ a \rightarrow b, c \rightarrow a, d \rightarrow a, d \rightarrow \neg b \}$$

Donnons-nous comme objectif de modéliser la connaissance modale suivante : la croyance $a \rightarrow b$ « *n'est pas toujours vraie* ». Par exemple, nous souhaitons qu'elle soit vraie pour c (permettant, depuis c , de produire b), et non valide pour d (d produit $\neg b$ et ne doit pas produire b). Considérons pour cela l'ensemble suivant :

$$E_{L_c} = \{ \quad p_{1.2} \rightarrow (a \rightarrow b), p_{2.2} \rightarrow (c \rightarrow a), p_{3.2} \rightarrow (d \rightarrow a), p_{4.2} \rightarrow (d \rightarrow \neg b), \\ p_{5.3} \rightarrow p_{1.2}, p_{6.3} \rightarrow \neg p_{1.2} \}$$

Cette proposition de transformation nécessite quelques commentaires :

- il existe de nombreuses manières de modéliser un ensemble de connaissances dans L_c . Cette remarque pose la question, ouverte, de l'apprentissage. Nous présentons celle qui nous semble la plus pertinente compte tenu de nos études à date,
- E_{L_c} respecte la syntaxe de L_p . Les propositions que nous notons $p_{i,j}$ portent les pensées. Par convention d'écriture, i singularise la proposition et j indique le rang de la pensée,
- nous n'avons pas utilisé le rang 1 des pensées parce qu'il n'est pas nécessaire pour modéliser cette connaissance modale. Nous présenterons au paragraphe suivant quelle utilisation nous lui réservons,
- selon le postulat contextuel, les pensées sont des propositions « *qui ne se prononcent pas* » : le « langage naturel » ne s'exprime directement qu'au travers des propositions

de L_p (c'est-à-dire de rang 0). C'est pourquoi la translation de E_{Lp} à E_{Lc} suit un algorithme qui propose une transformation automatisable,

- les deux dernières formules portent la connaissance modale que nous souhaitons exprimer. La modélisation est générique : elle ne contient ni c ni d . Le sens qu'on peut donner à $p_{5.3}$ est que $a \rightarrow b$ est vraie, et à $p_{6.3}$ que $a \rightarrow b$ « n'est pas ».

L'interprétation sémantique produit $\{a, b\}$ pour c , et $\{a, \neg b\}$ pour d :

- les contextes maximaux identifient les combinaisons de pensées autorisées par les connaissances modales. Dans l'exemple, on obtient $\{p_{5.3}\}$, qui porte $a \rightarrow b$, et $\{p_{6.3}\}$, qui ne porte pas $a \rightarrow b$,
- chaque contexte maximal est ensuite enrichi du contexte crédible. Or $p_{3.2}$ et $p_{4.2}$ portent une incohérence sur $p_{1.2}$ pour d , puisque $\{p_{1.2}, p_{3.2}, p_{4.2}, d\} \vdash_{Lp} b \wedge \neg b$. Ils sont donc rejetés des modèles vérifiant $\{p_{5.3}, d\}$, mais se retrouveront dans $\{p_{6.3}, d\}$.

En exemple d'application, cela permet de modéliser :

- que les hirondelles (notées c) volent (noté b), parce que les hirondelles sont des oiseaux (notés a , donc $c \rightarrow a$) et que les oiseaux volent (c'est-à-dire $a \rightarrow b$),
- et que les autruches (notées d) sont des oiseaux ($d \rightarrow a$) qui ne volent pas ($d \rightarrow \neg b$). La connaissance que les oiseaux volent ($a \rightarrow b$) est écartée des contextes portant des autruches du fait de la contradiction $b \wedge \neg b$ qu'elle engendre.

Les autruches héritent de l'ensemble des caractéristiques associées aux oiseaux à l'exception de l'attribut « voler ». Et, sauf connaissance explicite contraire, tout autre type d'oiseau (un moineau par exemple) sera traité comme les hirondelles, et associé aux mêmes attributs.

L'opération étant fastidieuse, nous avons informatisé le processus pour pallier cette difficulté. Il permet d'intégrer plusieurs connaissances modales, simples ou complexes, pour vérifier le comportement de la théorie au-delà du cas d'école présenté.

Modélisation de connaissances prédictives

Enrichissons à présent notre exemple initial en supposant dans L_p l'ensemble de propositions $\{a, b, c, d, e, f\}$ et la théorie suivante :

$$E_{L_p} = \{ a \rightarrow b, c \rightarrow a, d \rightarrow a, d \rightarrow \neg b, e \rightarrow (c \rightarrow f) \}$$

La formule $e \rightarrow (c \rightarrow f)$ est syntaxiquement équivalente à $c \rightarrow (e \rightarrow f)$. Cette symétrie ne permet pas d'associer f à e ou à c . C'est notamment ce qui a conduit à la constitution de la logique des prédicats. Or nous souhaitons modéliser que f est un prédicat de e , et c un complément descriptif du prédicat – pour dire, par exemple, « le chat (noté e) attaque (noté f) l'hirondelle (notée c) ». Considérons pour cela l'ensemble suivant dans L_c :

$$E_{L_c} = \{ \begin{array}{l} p_{1.2} \rightarrow (a \rightarrow b), p_{2.2} \rightarrow (c \rightarrow a), p_{3.2} \rightarrow (d \rightarrow a), p_{4.2} \rightarrow (d \rightarrow \neg b), \\ p_{5.3} \rightarrow p_{1.2}, p_{6.3} \rightarrow \neg p_{1.2}, \\ p_{7.1} \rightarrow (c \rightarrow f), p_{8.2} \rightarrow (e \rightarrow p_{7.1}), p_{9.3} \rightarrow p_{8.2}, p_{10.3} \rightarrow \neg p_{8.2} \end{array} \}$$

$p_{7.1}$, $p_{8.2}$, $p_{9.3}$ et $p_{10.3}$ portent la connaissance prédicative. Elles génèrent une asymétrie, en associant f à e sous c . L'interprétation sémantique se construit en plusieurs étapes :

- une étape interprétant séparément e et c . Elle va, en utilisant $p_{8.2}$, associer $c \rightarrow f$ (c'est-à-dire $p_{7.1}$) à e , pour exprimer : $c \rightarrow f$ est vraie dans les modèles vérifiant e ,
- la seconde étape consiste à croiser les sémantiques des deux perspectives e et c . La résultante porte la proposition f , et conserve $p_{8.2}$ qui trace la relation entre $c \rightarrow f$ et e .

La relation sémantique entre la fonction que nous venons de présenter et le langage naturel a des similitudes avec la logique des relations de Jean Piaget (1949). Nous développerons ce point dans une prochaine publication.

L'outil évoqué précédemment nous a permis de tester la méthode en introduisant plusieurs connaissances prédicatives, croisées voir même contradictoires entre elles, pour s'assurer pratiquement des comportements annoncés par la théorie.

Nos travaux précédents (cf. Arnaud Kohler, 2019) portaient sur les relations entre L_c , L_p et ses extensions modales. L'interprétation sémantique que nous venons de décrire étend la

consommation des contextes épistémiques par une fonction récursive, établissant ses liens avec la logique des prédicats.

Une explication : le comportement sémantique de la négation

De manière générale, les logiques non contextuelles modélisent que « *a est vraie* » ou que « *a est fausse* ». Dans L_c , ces deux expressions se lisent : « *Je pense que a est vraie* » (sous-entendu : selon un contexte) et « *Je pense que a est fausse* » (sous-entendu : selon un autre contexte). Il est dès lors possible de modéliser « *Je pense que a n'est pas vraie* » et « *Je pense que a n'est pas fausse* », qui ne disent pas tout à fait la même chose, ou « *Je ne pense pas que a soit vraie* », qui exprime une information encore différente.

L_c propose une alternative au problème de la double négation, en permettant d'exprimer « *Je pense que a n'est pas fausse* » et, dans le même temps, « *Je ne pense pas que a soit vraie* » - c'est à dire « *je pense ($\neg \neg a$) et je ne pense pas (a)* ».

C'est cette expressivité qui permet de capturer les modalités épistémiques et les prédicats en restant dans la syntaxe de la logique propositionnelle, pour exprimer par exemple que « *A aime B* » et, dans le même temps, que « *B n'aime pas A* » - ou de ne rien dire concernant *B*.

Ecart entre la logique des prédicats et la logique contextuelle

L'écart premier entre la logique des prédicats et la logique contextuelle est que cette dernière n'a pas de quantificateur. La formule « *Quel que soit x , $f(x)$* » n'est donc pas capturable dans L_c , et devient $c \rightarrow f.f$ est dès lors interprétée comme crédible tant que rien ne dit qu'elle ne l'est pas. Les deux formalismes modélisent des conceptions différentes du raisonnement : un comportement démonstratif pour l'un, et faillibiliste pour l'autre.

Conclusion

Sous leur apparente simplicité, les 3 principes d'Aristote (tiers exclu, non contradiction et identité) cachent une forte complexité d'utilisation, et une capacité d'expression qui n'a peut-être pas encore été complètement explorée.

Ainsi, la logique contextuelle respecte strictement les règles syntaxiques monotones de la logique propositionnelle. Pourtant, en modélisant la réflexivité du raisonnement dans la syntaxe et en étendant la sémantique par une fonction récursive, nous associons à L_p une fonction d'interprétation non monotone qui permet l'exploitation de connaissances incohérentes, modales épistémiques ou prédicatives. Ce résultat ne préjuge pas de ce qu'est l'Intelligence (cf. Luc Julia, 2019), mais il ouvre des voies de recherches pour certaines difficultés rencontrées en Intelligence Artificielle symbolique.

Dans le cadre de ces travaux, un piège consiste à biaiser les études en sélectionnant involontairement les inférences qui permettent d'obtenir les conclusions souhaitées. Il est facile d'y succomber dans L_p . Dans L_c , la présence des pensées démultiplie les risques d'erreur. Pour y pallier, nous avons développé un outil qui reproduit les fonctions de la logique contextuelle. Il nous a permis de vérifier les conclusions annoncées par la théorie - sur l'exemple présenté et sur des cas plus complexes. L_c garantit la consistance syntaxique de toute théorie, ce qui résout la première étape de la révision des croyances. Nous avons profité de cette propriété pour implémenter des fonctions d'apprentissage, et permettre au système d'absorber n'importe quelle connaissance.

L'outil est adossé aux algorithmes de la logique propositionnelle. Ils subissent des temps de calcul qui ne sont pas acceptables dans le cadre d'une IA. Les propriétés de L_c semblent bien se marier avec les résultats constatés empiriquement par les sciences cognitives (cf. Arnaud Kohler, 2019). C'est cette voie que nous explorons actuellement pour tenter de remédier à ce problème, en suivant la piste initiée par les seuils cognitifs. Nous étudions en parallèle le sujet des langages naturels, en revisitant la fonction d'interprétation sémantique de L_c au travers de la logique des relations proposée par Jean Piaget (1949).

Références bibliographiques

- ANDLER, Daniel, *Introduction aux sciences cognitives*, Gallimard, 2004
- JULIA, Luc, *L'intelligence artificielle n'existe pas*, Broché, 2019
- KOHLER, Arnaud, *La logique contextuelle*, réf. hal-02120285, 2019
- PIAGET, Jean, *Traité de logique : essai de logistique opératoire*, A. Colin, 1949