Résultats de théorie abstraite des modèles dans le cadre des institutions : vers la combinaison de logiques.

Fabrice Barbier

THÈSE

présentée par

Fabrice Barbier

pour obtenir le titre de

DOCTEUR de L'UNIVERSITÉ D'ÉVRY VAL D'ESSONNE

Spécialité : **Informatique**

Résultats de théorie abstraite des modèles dans le cadre des institutions : vers la combinaison de logiques

Date de soutenance : 5 décembre 2005

Composition du jury

Rapporteurs	M. Gilles DOWEK
	M. Pierre-Yves SCHOBBENS
Examinateurs	M. Marc AIGUIER
	M. Richard LASSAIGNE
	M. Benjamin WERNER
	M. Răzvan DIACONESCU
Directrice de thèse	Mme Pascale LE GALL

Thèse préparée au Laboratoire de Méthodes Informatiques
de l'Université d'Évry Val d'Essonne - UMR 8042 du CNRS

Résumé

Dans le cadre du génie logiciel formel, la structuration et la modularité des spécifications axiomatiques sont des propriétés indispensables pour le développement de logiciels de qualité. L'étude logique-indépendante de ces propriétés a mis en évidence le caractère suffisant de la propriété d'*interpolation de Craig* pour certains aspects de la structuration et la modularité. Le but de cette thèse est alors double :

1. En théorie standard des modèles[1], l'interpolation de Craig est équivalente à la propriété de *consistance de Robinson*. Dans [114], A. Salibra et G. Scollo ont généralisé cette équivalence dans le cadre abstrait des institutions possédant la négation. Dans la pratique cependant, un tel résultat reste insuffisant car la consistance de Robinson repose sur une propriété souvent très difficile à établir : l'*équivalence élémentaire* de modèles. La *méthode des diagrammes* est une méthode de théorie standard des modèles permettant justement de construire des modèles élémentairement équivalents. De façon à pouvoir démontrer la consistance de Robinson dans un cadre général, nous généralisons cette méthode, ainsi que la notion de morphisme élémentaire sous-jacente, dans une extension du cadre des institutions appelé institution stratifiée.
2. Une propriété sous-jacente à la consistance de Robinson est le lemme de l'union des chaînes de Tarski. Pour la généralisation de ce dernier résultat (et donc de la consistance de Robinson), nous serons amenés à étendre les institutions stratifiées en explicitant les connecteurs logiques. Ceci nous amènera alors naturellement à étudier la notion de combinaison de logiques connue aussi sous le nom de fibration depuis les travaux pionniers de D. Gabbay (*cf.* [58]). Nous généralisons alors cette notion de fibration afin de mieux prendre en compte la pratique de la définition d'un formalisme en théorie de la spécification. En effet, lors de la définition d'un nouveau formalisme de spécification, ce dernier n'est souvent pas la simple union de formalismes existants mais apporte des modifications aux formalismes sous-jacents. Ces modifications seront représentées par des co-morphismes d'institutions. Ne perdant pas de vue le problème de la structuration et de la modularité des spécifications, nous nous intéressons à la préservation de l'interpolation de Craig et de la consistance de Robinson au travers de cette nouvelle définition de la combinaison de logiques.

1. On appelle théorie standard des modèles l'étude des modèles de la logique des prédicats du premier ordre avec égalité, également appelée logique du premier ordre classique.

Table des matières

Introduction

Cadre de travail

Depuis la fin des années soixante et la crise du logiciel, il est clair que les systèmes informatiques réels sont bien trop imposants et complexes pour être compris dans leur intégralité par un unique concepteur. Ce constat induit la nécessité de suivre une méthode rigoureuse de développement si on veut que les logiciels aient une chance d'être livrés à temps, de correspondre à leur cahier des charges et d'être faciles à maintenir et à faire évoluer. Ceci implique non seulement de définir des critères pour évaluer la qualité des logiciels produits mais aussi des méthodes et techniques pour développer des logiciels qui satisferont à ces critères. La définition de tels critères, méthodes et techniques requiert l'étude des caractéristiques propres des systèmes informatiques et de leur processus de développement. Ceci caractérise ce que l'on appelle couramment, par analogie avec les autres corps d'ingénierie, le *génie logiciel*, *i.e.* la donnée d'un *corpus* de méthodes plus ou moins rigoureuses et clairement identifiées par tous permettant à une (ou plusieurs) équipe(s) de travailler séparément sur différentes parties d'un même projet et d'assurer, au moyen d'un certain nombre de critères prédéfinis, la qualité du produit livré.

Le travail présenté ici s'inscrit dans le cadre du *génie logiciel formel*. Celui-ci se caractérise par l'utilisation de toute *méthode formelle* permettant de renforcer la « qualité » des systèmes informatiques. Par qualité des systèmes informatiques, nous entendons un bon fonctionnement dudit système selon une référence de correction disponible représentée par un *cahier des charges*, également appelé *expression des besoins* ou encore *spécification*. Dans le cadre du génie logiciel formel, ce cahier des charges doit être décrit au moyen de méthodes formelles, *i.e.* un ensemble de techniques mathématiquement fondées et vérifiables par un ordinateur. Ceci permet alors d'aborder le problème de la qualité de façon scientifique, car il devient possible de montrer rigoureusement que les composants d'un logiciel décrits au moyen de telles méthodes répondent logiquement à leur spécification. Les méthodes formelles se différencient ainsi des méthodes de conception informelles (UML [102], Merise [128], etc.) par la donnée d'une syntaxe et d'une sémantique rigoureusement définies et non ambiguës permettant, de ce fait, de prouver mathématiquement le bon fonctionnement d'un système et d'en évaluer la qualité selon des critères stricts. En fait, bien que les méthodes informelles permettent de donner un schéma général du futur système et fournissent ainsi une aide non négligeable pour le développement de larges applications informatiques, certains aspects du système spécifié restent exprimés en langage naturel. Or, parce qu'ils ne répondent pas à des règles de construction rigoureuses et que leurs productions syntaxiques sont sujettes à de multiples interprétations, les langages naturels ne permettent pas d'éliminer toute ambiguïté du schéma du système et peuvent même générer des incompréhensions, voire gêner l'identification, et donc la gestion, d'erreurs de conception. Dans le cadre du développement de logiciels avec de fortes contraintes, tels que les systèmes critiques où des vies humaines sont en jeu, on ne peut donc pas se reposer sur ces méthodes informelles pour assurer du bon fonctionnement et de la qualité des systèmes informatiques.

Les méthodes formelles

Une *méthode formelle*, ou *formalisme de spécification*, est la donnée d'une théorie mathématique adaptée à la conception, la validation et la vérification de systèmes informatiques, *i.e.* une théorie munie de techniques formellement définies permettant de réaliser des logiciels corrects (au moins partiellement) à partir de leur spécification. Cette vue classique des formalismes de spécification amène à différencier (au moins) deux classes de méthodes formelles :

- les *méthodes orientées modèles* (ou *méthodes constructives*), telles que VDM, Z et B. Les concepteurs construisent un unique modèle à partir des structures de données et des primitives de construction offertes par le langage de spécification (*cf.* respectivement [70], [127] et [1]). Le logiciel produit est dit correct par rapport à sa spécification si les fonctions qu'il offre ont le même comportement que dans le modèle spécifié ;
- les *méthodes orientées propriétés* (ou *méthodes axiomatiques*, ou encore *méthodes déclaratives*). Les concepteurs définissent tout d'abord les éléments atomiques du langage (*i.e.* l'interface du problème considéré) si bien qu'il existe une infinité de modèles qui peuvent fournir une interprétation sémantique pour chacun de ces éléments. Ensuite, les concepteurs établissent un certain nombre de propriétés (des formules souvent appelées *axiomes* car elles ne sont pas à prouver mais imposées) que le système doit vérifier. De tous les modèles susmentionnés, seuls quelques-uns satisfont lesdites propriétés, les autres sont écartés. Un logiciel est alors correct par rapport à sa spécification s'il correspond à un modèle vérifiant la spécification.

Parmi les autres classes de méthodes de spécification, on trouve les formalismes opérationnels[2]. À partir d'un ensemble d'actions élémentaires, ces méthodes de développement s'attachent à décrir les séquences d'actions autorisées du système. Les représentants les plus significatifs de ces méthodes sont les réseaux de Petri (*cf.* [108]) et les algèbres de processus (*cf.* [14]).

Maintenant, la taille et la complexité des systèmes informatiques actuels sont bien trop importantes pour que ces systèmes soient conçus, maîtrisés ou simplement étudiés par un unique concepteur. De ce fait, il est aujourd'hui nécessaire de disposer de méthodes permettant à la fois le développement pas-à-pas des logiciels à partir du cahier des charges et le « découpage » d'un

2. Les formalismes opérationnels peuvent être vus comme un sous-ensemble des méthodes orientées modèles dans la mesure où la démarche de conception est la même.

système en plusieurs spécifications (suivant les différents centres d'intérêts) suivi du « recollement » de l'ensemble (sans générer de problème supplémentaires). De même, afin d'assurer de la qualité des logiciels produits, il faut fournir les moyens de vérifier la cohérence des spécifications produites et leur correction par rapport au cahier des charges tout en tenant compte de cette structuration. Dans l'optique de développement de logiciels de qualité, la définition de telles méthodes et techniques est alors indispensable. Celles-ci se partagent en deux groupes différents suivant les problèmes qu'elles permettent de résoudre :

- **celles qui aident à la *conception* de systèmes informatiques.** Nous trouvons ici la *structuration* et le *raffinement* de spécifications (*cf.* [21, 116, 104]);
- **celles qui aident à la *validation/vérification* de systèmes informatiques.** Nous trouvons ici la *preuve*, le *test* et le *prototypage* de spécifications (*cf.* [90, 77, 9]).

La structuration et le raffinement répondent au problème de la taille des systèmes informatiques et donc des spécifications qui les définissent. Ces deux techniques sont utilisées lorsque la taille des spécifications et des systèmes dépasse l'entendement d'un unique concepteur. Elles constituent ainsi des aides non négligeables pour la conception de systèmes informatiques larges et complexes (ce qui est généralement le cas) et permettent d'en assurer la qualité au fur et à mesure du développement. Dans le cas de la structuration, il s'agit de définir le comportement global d'un système à partir d'un ensemble de comportements locaux (définissant des sous-systèmes) hiérarchisés par une relation d'utilisation. Cela se traduit par l'emploi d'opérateurs sur les spécifications qui permettent par exemple de faire l'union de deux spécifications ou encore d'importer ou d'exporter des spécifications entre modules. Dans le cas du raffinement, il s'agit de faire, à partir d'une spécification initiale, un choix de conception parmi tous ceux qui sont licites au sens de la spécification initiale. Cela se traduit naturellement par une restriction de la classe de modèles en fonction des choix de conception effectués. De ce fait, le raffinement est souvent qualifié de structuration verticale par opposition à la structuration horizontale des spécifications décrite ci-dessus. Lorsque ces deux techniques sont utilisées conjointement, on parle alors de structuration transversale.
La preuve et le test répondent quant à eux au problème de correspondance du produit fini avec le produit commandé, *i.e.* ils permettent de vérifier *a posteriori* la cohérence du logiciel produit et d'assurer que son comportement est conforme à celui décrit dans le cahier des charges. Dès que l'on utilise une méthode formelle, il apparaît naturel de s'intéresser à la preuve car celle-ci est une technique mathématiquement fondée. De par la définition d'une logique, les preuves peuvent aussi bien être sémantiques que syntaxiques. Dans le cadre du génie logiciel formel, c'est usuellement l'approche syntaxique qui est choisie car les preuves s'y résument à de simples manipulations syntaxiques suivant les règles définies par le calcul de la logique considérée. Elles sont donc mécaniquement vérifiables[3]. En pratique, l'utilisation des diverses techniques de preuves

3. Notons que du côté sémantique, il existe, pour certains systèmes logiques comme la logique propositionnelle et certaines de ses extensions modales, des techniques automatisables de preuve sémantique qui confrontent une abstraction d'une réalisation, représentée par un modèle mathématique (traditionnellement un automate), à sa spécification. C'est ce qu'on appelle le *model-checking*, ou *vérification de modèles* (*cf.* [69]).

est limitée par leur difficulté à traiter des spécifications de tailles réelles en des temps humainement raisonnables. Seules les parties les plus critiques sont sujettes à de la preuve et la correction des logiciels n'est donc que partiellement assurée. De plus, un logiciel s'exécute au sein d'un environnement qui n'est pas pris en compte par la spécification du système et est donc implicitement considéré comme correct (c'est le cas des aspects matériels ou du compilateur). À la différence de la preuve, le test permet d'étudier la correction d'un logiciel intégré dans son environnement. Il est à rapprocher de la vérification de modèles dans le sens où il consiste en la confrontation d'une réalisation du système sous test à sa spécification. La seule différence est que cette confrontation ne concerne que quelques cas jugés pertinents, *i.e.* soupçonnés de mettre en défaut la réalisation. Contrairement à la preuve, le test n'a pas pour but de valider une réalisation mais plutôt de trouver un contre-exemple à son bon fonctionnement et renforcer ainsi le niveau de confiance des utilisateurs sur la correction de la réalisation. De même que la preuve, le test ne permet donc d'assurer qu'une correction partielle d'un logiciel. Néanmoins ces deux techniques de vérification se combinent très bien dans la pratique et permettent de produire des logiciels de qualité. Enfin, le prototypage de spécifications regroupe l'ensemble des techniques qui permettent, à partir d'une spécification, d'engendrer une réalisation correcte et exécutable. En général, les réalisations engendrées par prototypage sont de trop faible efficacité pour représenter la réalisation finale du système. Elles sont le plus souvent utilisées soit comme des spécifications dont la théorie sous-jacente est décidable, permettant ainsi de prouver automatiquement des propriétés attendues mais non explicites dans la spécification (c'est le cas des systèmes de réécriture), soit comme référence exécutable de la réalisation finale (pour répondre au problème de l'oracle dans le cadre du test fonctionnel par exemple). Le prototypage permet donc de vérifier les logiciels en cours de développement, en plus d'être une méthode de démonstration.
La figure ci-dessous montre le processus de développement de logiciels en partant du cahier des charges fourni par les utilisateurs. Ce dernier permet d'écrire un ensemble de spécifications structurées (SP) qui sont ensuite raffinées (SP'), *i.e.* SP' est obtenue à partir de SP en faisant certains choix d'implantation. À partir des spécfications ainsi obtenues, une selection permet de soumettre un jeu de test dont le résultat est confronté à un oracle. À partir de ces mêmes spécifications et avec l'aide des techniques de réécriture, il est également possible de produire un prototype (spécification exécutable) du système. Enfin, les techniques de preuve permettent d'assurer, à chaque étape de raffinement, que la spécification SP' est bien une réalisation correct de la spécification SP.

Les méthodes axiomatiques

Dans le cadre de cette thèse, nous nous intéressons aux spécifications de type axiomatique. Le fait de vouloir utiliser des méthodes axiomatiques pour la spécification formelle de logiciels a les conséquences suivantes :

- **il faut expliciter les propriétés que l'on veut établir par ces méthodes axiomatiques.** Pour que le procédé de validation de ces propriétés soit vérifiable par un ordinateur, il faut que leur énoncé lui-même soit mis sous une forme directement « compréhensible » par un ordinateur,

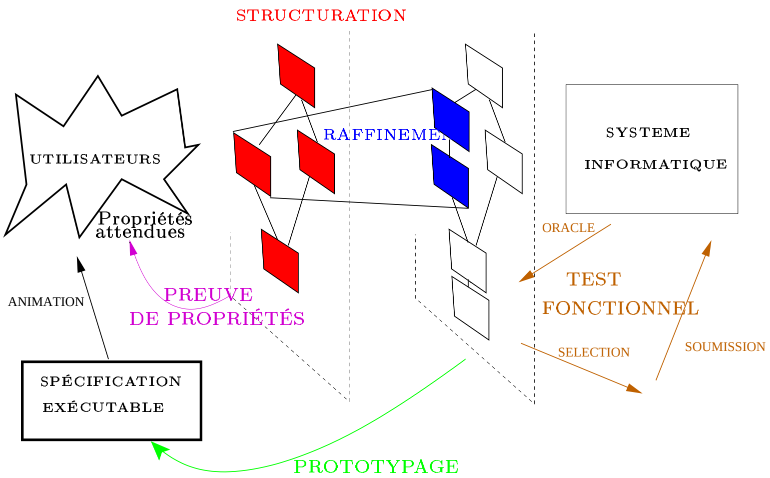

FIG. 1 – *Processus de développement de logiciels*

i.e. sous forme symbolique. La définition des suites de symboles reconnaissables pour une méthode axiomatique donnée en caractérise la *syntaxe* et l'énoncé d'une propriété du système respectant cette syntaxe est usuellement appelé une *formule*. La collection de tous les énoncés à établir sur le système considéré en constitue la *spécification* ;

– l'énoncé symbolique d'une propriété ne suffit pas à lui seul, puisqu'il n'est qu'une suite de caractères et de symboles. **Il faut lui donner un sens « dans le monde réel » correspondant au phénomène que l'on veut garantir par cette propriété.** Cela conduit nécessairement à représenter le monde réel (celui dans lequel le système existe) sous une forme rigoureuse, c'est-à-dire un ensemble de modèles mathématiques, afin de faire porter formellement cette propriété sur des objets précis. Cette partie des méthodes axiomatiques s'appelle la *sémantique* ;

– enfin, pour appliquer les méthodes axiomatiques à la correction de logiciel, la question qui se pose alors est : « *Comment aboutir par ordinateur à démontrer la suite de symboles qu'est une propriété ?* ». La seule chose que sache faire un ordinateur étant justement de manipuler des suites de symboles, **il faut alors définir les conditions de manipulation des formules et comment elles peuvent conduire à l'obtention d'une propriété.** On parle alors de *système formel*, de *système d'inférence* ou encore de *calcul*.

Schématiquement, ce lien entre les spécifications axiomatiques et le processus de développement de logiciels se traduit de la façon suivante :

Ces trois aspects des méthodes axiomatiques (syntaxe, sémantique, calcul) définissent les éléments de base de ce qu'on appelle une *logique*, ou *système logique*. Quelque soit la logique considérée, ces trois éléments sont bien entendu loin d'être indépendants comme l'indique le schéma simplifié de la figure 2. La syntaxe définit le domaine dans lequel travaille la sémantique et le calcul en fixant le langage, *i.e.* les symboles utilisés et les constructions autorisées. La sémantique quant à elle doit bien évidemment être capable de donner un sens à toutes les constructions définies par la syntaxe. Enfin, le calcul n'étant qu'une

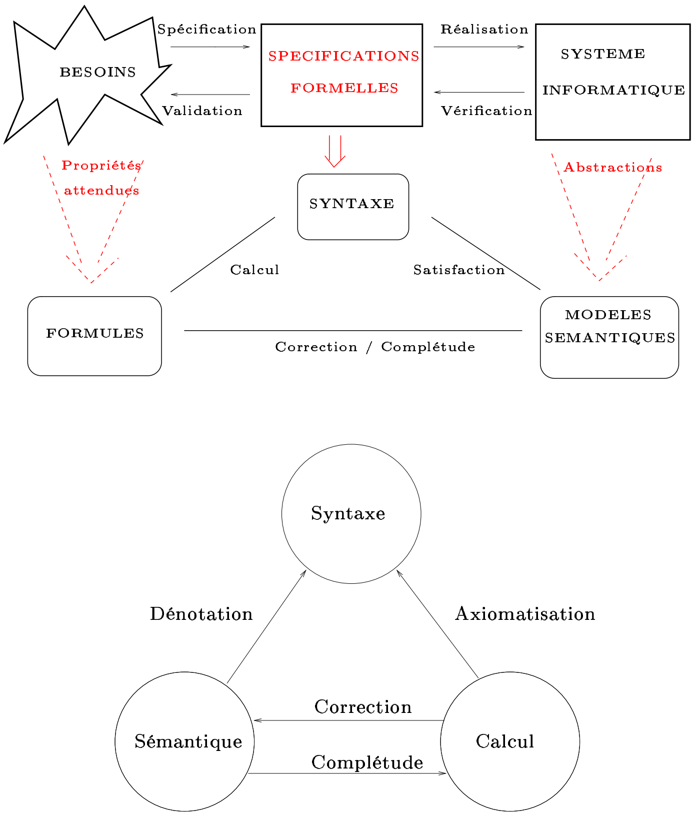

FIG. 2 – *Système logique*

manipulation symbolique *a priori* sans aucun sens, il est indispensable de s'assurer que ce dernier représente aussi fidèlement que possible le « monde réel » (en tout cas la partie du monde réel visé). Faute de pouvoir mixer le monde réel et les manipulations syntaxiques, on demande tout naturellement que ces manipulations symboliques soient en accord avec la vision mathématique du monde réel choisie, *i.e.* la sémantique. On parle alors de *correction du calcul*. Lorsque qu'elle traduit totalement cette vision mathématique, on parle de *complétude*. Bien entendu, la première propriété est absolument nécessaire, car proposer un calcul sans définir en quel sens il est correct (*i.e.* ne pas fournir de sémantique) enlève de fait toute crédibilité aux méthodes formelles. C'est seulement une fois établie la correction des règles du calcul par rapport à la sémantique que l'on peut faire confiance à un calcul formel. Seulement alors on peut s'appuyer sur une compréhension intuitive de la sémantique pour guider les preuves formelles. La complétude en revanche est seulement souhaitable car, s'il est intéressant de traduire totalement la sémantique par des règles de calcul, cela n'est pas toujours possible (une conséquence de l'incomplétude de Gödel). Bien entendu, on peut opposer à ce qui précède que la sémantique est elle-même définie par un système formel (usuellement celui défini par les axiomes de Zermelo-Fraenkel avec axiome du choix (ZFC) pour la théorie des ensembles (*cf.* [126, 33])) dont nous ne pouvons pas montrer la cohérence. Cependant, ce dernier système possède un degré de confiance important de par son ancienneté d'utilisation (il est utilisé depuis presque un siècle sans qu'aucune incohérence n'ait été établie). On comprend alors mal pourquoi on ne fait pas l'économie de la définition d'un nouveau système en utilisant directement le système ZFC. La raison en est que le système formel fournit par ZFC est beaucoup trop riche, *i.e.* le pouvoir d'expression de ZFC est largement supérieur à ce qui est nécessaire pour la spécification de systèmes informatiques, aussi larges et complexes soient-ils. La conséquence de ce trop fort pouvoir d'expression de ZFC est que les preuves effectuées dans ce système sont difficilement exploitables. En effet, des caractéristiques propres au domaine en question doivent être démontrées comme des théorèmes dans ZFC (le remplacement d'égal par égal pour la logique équationnelle par exemple) ce qui alourdit les preuves et les rend peu lisibles. Au contraire, la définition d'un système formel adapté au problème donné (le fait de ne considérer que des fonctions totales par exemple) allège les preuves et permet de mieux se concentrer sur les points importants de ces dernières en faisant abstraction des tautologies du domaine (comme par exemple le fait qu'une fonction totale soit définie pour toutes les valeurs de l'espace de valeurs considéré).

Multitude des logiques

Que ce soit en mathématiques ou en informatique, il existe aujourd'hui une multitude de logiques répondant à ce découpage syntaxe, sémantique et calcul. Chacune d'entre elles a été créée dans un but précis (par exemple modélisation des individus, du temps, de la croyance, des types de données complexes, *etc.* ou encore quantification des variables, des fonctions, des ensembles, *etc.*) et est dédié à un domaine d'application particulier (évolution temporelle d'une base de données, comportement d'un système au cours du temps, modélisation de l'intelligence, fondement d'objets théoriques, *etc.*). Dans la pratique, cette diversité est absolument nécessaire pour répondre aux besoins particuliers des différents utilisateurs de formalismes de spécification : logiciens, informaticiens, philosophes, linguistes, *etc.* De plus, pour chaque logique, et suivant les goûts et besoins spécifiques des utilisateurs, différents formalismes de spécification peuvent être définis. Ils différeront alors par leurs notations et les outils et techniques définis.

En mathématiques, les logiques sont utilisées pour donner un fondement aux objets usuellement manipulés par les mathématiciens (tels les groupes, les treillis, *etc.*). Les spécifications écrites sont rarement utilisées par la suite (ce sont donc bien des logiques et non des formalismes de spécification qui sont utilisées). Seules les trois exigences suivantes sont pertinentes dans le cadre du travail des mathématiciens :

- non contradiction des axiomes ;
- complétude syntaxique[4] (*i.e.* pour toute propriété, elle-même ou sa négation est prouvable) ;
- indépendance des axiomes.

L'introduction d'un nouveau formalisme découle alors de l'impossibilité pour les logiques existantes d'exprimer certaines propriétés intéressantes, *i.e.* leur

4. La complétude syntaxique n'est pas à confondre avec la complétude sémantique qui est une propriété établie sur une logique, et montre l'équivalence entre le calcul et la sémantique.

pouvoir d'expression n'est pas suffisant. C'est le cas par exemple de la logique des prédicats du premier ordre dont le pouvoir d'expression est insuffisant pour caractériser des ensembles de cardinalité infinie donnée ou encore pour spécifier certaines structures algébriques tels les ensembles bien-fondés[5]. De la même manière, la description de l'induction mathématique demande une quantification sur les ensembles et non plus sur les valeurs. Cette propriété n'est pas exprimable dans le cadre de la logique classique des prédicats du premier ordre à moins de considérer une infinité d'axiomes (un par formule) dont voici le schéma :

$$(\varphi[0] \wedge (\forall n,\ \varphi[n] \Rightarrow \varphi[n+1])) \Rightarrow \forall x,\ \varphi[x]$$

où φ est une formule du premier ordre quelconque possédant x comme variable libre.

En informatique, et plus précisément dans le domaine du génie logiciel formel, c'est rarement le pouvoir d'expression qui est en cause dans la définition d'un nouveau formalisme. Ce qui amène à introduire un nouveau formalisme de spécification pour la description de systèmes informatiques est l'inadaptation des formalismes existants au *domaine d'application* visé en termes de concision des spécifications résultantes, ainsi qu'en termes de clarté, lisibilité et facilité d'utilisation des spécifications tout au long du processus de développement (structuration, raffinement, preuve, test, prototypage, *etc.*). C'est ce que l'on nomme le *pouvoir d'expressivité*. En effet, du point de vue de la simple description (*i.e.* indépendamment de l'exploitation des spécifications pour établir leur validation par exemple), la logique des prédicats du premier ordre est souvent suffisante pour la spécification des systèmes informatiques. D'ailleurs, la plupart des formalismes utilisés dans le cadre du génie logiciel formel se traduisent assez aisément dans cette dernière. Par exemple, toute spécification écrite avec la logique équationnelle multi-sortes se traduit dans la logique classique des prédicats du premier ordre en représentant chaque type par un prédicat unaire et en explicitant par une multitude d'axiomes, le domaine de définition de chacune des fonctions du langage. La spécification ainsi obtenue est moins concise du fait de cette description explicite (*i.e.* par axiome) du typage. Il va sans dire que cette spécification sera beaucoup plus difficile à exploiter que celle (équivalente) écrite avec la logique équationnelle multi-sortes. Il en va de même pour les logiques modales où les spécifications du premier ordre correspondantes sont alourdies par la description explicite des états (en tant qu'éléments d'un ensemble) et de la relation d'accessibilité (une relation binaire sur l'ensemble des états).
Ce pouvoir d'expressivité se retrouve au niveau des langages de programmation. En effet, tous les langages de programmation possèdent le même pouvoir d'expression établi par la théorie de la calculabilité (si l'on accepte la thèse de Church-Turing). Cependant, l'utilisation préférentielle d'un langage par rapport à un autre dépend des critères de qualité choisis sur les programmes écrits (rapidité d'exécution, facilité de la maintenance et/ou de la validation formelle de programmes, *etc.*.). Le langage Caml, par exemple, est bien adapté à l'écriture

5. Ces résultats sont des conséquences simples d'un résultat issu de la théorie standard des modèles connu sous le nom de *théorème de Löweineim-Skolem* (*cf.* [33]). Ce théorème établit que, dans le cadre de la logique des prédicats du premier ordre, si une formule possède un modèle de domaine infini alors elle possède un modèle de domaine infini pour toute cardinalité infinie.

de programmes de haut niveau car il offre des constructions syntaxiques abstraites. À l'inverse, le langage C offre une plus grande souplesse pour qui veut contrôler de près les accès mémoire en fournissant des constructions syntaxiques plus concrètes et plus proches du comportement réel de la machine.

Ainsi, au même titre que les logiciens modélisent les mathématiques et les physiciens ou les biologistes modélisent la nature, les informaticiens dans le domaine du génie logiciel formel modélisent les méthodes existantes ou à découvrir de conception des systèmes informatiques (par objet, par aspect, modulaire, *etc.*.). La seule différence avec nos collègues logiciens et physiciens (la description des objets biologiques étant encore très peu formalisée) est que notre objet d'étude maximal n'est plus à découvrir étant donné que nous n'irons pas plus loin que ce qui est calculable. Ainsi, comme nous venons de l'expliquer ci-dessus, la définition d'un nouveau formalisme (ceci est aussi vrai pour la définition d'un nouveau paradigme de programmation) n'est pas l'impossibilité des anciennes théories (représentées ici par des logiques) de prendre en compte une observation quantitative sur le sujet d'étude, mais plutôt leur mauvaise adaptation pour traiter ce sujet d'étude (observation qualitative).

Besoin d'un méta-cadre

Du fait du développement phénoménal de l'informatique, les besoins en logiciels n'ont cessé d'augmenter et de se diversifier. On a alors vu l'apparition de nombreux langages de programmation et formalismes de spécification résultant à la fois d'une meilleure adéquation aux besoins, d'investigations théoriques et d'expérimentations. La contrepartie à ce développement tous azimuts des langages de programmation et de spécification est l'accumulation d'une formidable quantité de travail due à la nécessaire redéfinition pour chaque langage, des techniques d'aide au dévelopement décrites plus haut (structuration, raffinement, preuve, test, prototypage).

Que ce soit en mathématiques ou en informatique, l'introduction d'un nouveau formalisme se fait toujours par la donnée de sa syntaxe, de sa sémantique et de son calcul. La validité des résultats classiques (décidabilité, correction et complétude du calcul, *etc.*) est ensuite étudiée. Enfin, on définit des techniques et outils d'aide à la conception (opérateurs de structuration, prouveurs de théorèmes, prototypeurs, générateurs de jeux de test, *etc.*). Or, même si la définition de ces différents éléments est propre à chaque langage, de nombreux points communs existent traduisant l'essence même de la création de formalismes de spécification. Ainsi qu'il est usuel de le faire en sciences (au moins celles qui ont reçu une axiomatisation mathématique), il est alors important d'essayer de faciliter la description (et donc l'introduction) de toute logique en généralisant cette démarche. L'intérêt d'une telle généralisation est d'éviter aux créateurs de logiques de développer de nombreux résultats formels qui ne sont souvent que des adaptations de résultats bien connus à leur nouveau formalisme. On applique alors l'aphorisme « *abstraire, c'est simplifier* » en faisant porter l'abstraction sur les formalismes de spécification. Ceci permet d'étudier les systèmes logiques de façon générale sans se soucier des détails propres à une logique particulière. Un tel *méta-cadre* autorise alors l'étude « logique-indépendante » des relations entre syntaxe, sémantique et calcul d'une part et, d'autre part, des relations entre ce triplet et les différentes techniques de développement. L'avantage de cette approche « méta » est que seule la méthode est explicite, *i.e.* on s'intéresse uniquement à ce qui constitute l'essence du problème considéré, comme par exemple le problème de la structuration des spécifications, indépendamment de sa réalisation effective dans un système logique particulier. Parce qu'elle permet de se focaliser sur les conditions essentielles, l'abstraction offre une vision générale sur l'art de concevoir une théorie, quel que soit la logique considérée. La *théorie des catégories* (*cf.* [10, 78]), en fournissant un tel méta-cadre, a permis quelques avancées dans ce domaine (*cf.* [71, 107]) comme par exemple la traduction de la logique équationnelle par F. Lawvere et plus généralement l'interprétation de la logique à partir de la théorie des topoi (*cf.* [64, 75, 76]). Malheureusement, la théorie des catégories ne permet pas de dégager aisément des résultats spécifiques aux formalismes de spécification. En effet, cette théorie a été introduite dans le but de donner aux mathématiques un langage fondationnel plus général que la théorie des ensembles. La logique mathématique étant une branche des mathématiques, son interprétation catégorielle a principalement permis de généraliser la théorie des modèles sous-jacente à chaque logique. De ce fait, seule la partie sémantique des formalismes de spécification en a bénéficié. Il en est de même pour les *logiques abstraites* de J. Barwise (*cf.* [11]) qui ont permis de s'intéresser à la théorie des modèles indépendemment de la logique sous-jacente (la théorie des modèles a initialement pour sujet d'étude les modèles associés à la logique du premier ordre classique). Dans le cadre de cette *théorie abstraite des modèles*, un des premiers grands résultats lié au pouvoir d'expression, le théorème de Lindström (*cf.* [30, 55]), établit que la logique classique du premier ordre est la seule logique fermée par $\wedge$, $\neg$ et $\exists$ qui satisfasse à la fois les propriétés de compacité dénombrable (si un ensemble dénombrable de formules n'a aucun modèle alors il existe un sous-ensemble fini de l'ensemble initial de formules qui n'a aucun modèle), et de Löweinem-Skolem descendant (si une formule a un modèle de domaine infini alors elle a un modèle de domaine dénombrable).

Cependant, cette première généralisation des système logiques n'était pas satisfaisante dans le cadre de formalismes dédiés au génie logiciel. En effet, une logique abstraite au sens de J. Barwise est la simple donnée d'un couple $(\mathcal{L}, \models)$ où $\mathcal{L}$ est l'ensemble des formules que l'on peut définir et $\models$ la relation de satisfaction entre les modèles et les formules. Des propriétés sont bien entendu requises sur ce couple (les modèles ne peuvent satisfaire que des formules dont ils partagent le même langage, deux modèles isomorphes sont indiscernables via $\models$, *etc.* [*cf.* [30] p.127 pour plus de détails]). Maintenant, en pratique, le volet syntaxique d'une logique repose sur la construction inductive de formules à partir de symboles de fonctions, de prédicats, de connecteurs propositionnels, de quantificateurs, *etc.* On remarque également qu'une logique contient des « parties fixes » (connecteurs propositionnels, quantificateurs, définition inductive des termes et des formules, *etc.*) et des « parties utilisateurs » que l'on introduit dans le but de résoudre un problème donné. Ces dernières sont classiquement appelées des *signatures*. Elles caractérisent les éléments de base spécifiques à un problème donné. Elles représentent donc l'interface du problème traité. Dans le cadre d'un système informatique, cette interface est importante. En effet, c'est souvent la seule partie visible d'un système pour les utilisateurs. De plus, un

système informatique est amené à voir son comportement évoluer au travers de son cycle de vie. Cette évolution se caractérise souvent en premier lieu par une modification de son interface (par ajout de fonctionnalités par exemple) puis par une modification de son comportement (par exemple en ajoutant des propriétés pour caractériser les nouvelles fonctionnalités). Pour ces raisons, la notion de signature, ainsi qu'un moyen de comparer ces signatures au niveau syntaxique puis au niveau sémantique, doivent apparaître dans le cadre d'une description générale de logiques dédiées à l'informatique. Pour mieux appréhender les liens entre la logique et son application à l'informatique, la théorie des *institutions* (*cf.* [63]) a alors été introduite par G. Goguen et R. Burstall. Elle généralise les logiques abstraites de J. Barwise dans le sens où elle introduit la notion de signature, ne restreint pas le langage aux symboles logiques classiques ($\wedge, \vee, \neg, \Rightarrow, \forall, \exists$) et conserve simplement la propriété de renommage (appelée plus communément, dans le cadre des institutions, *condition de satisfaction*) pour son rôle important dans la structuration de spécifications. Une institution distingue au sein des théories les signatures des formules. Sans détailler la structure des modèles ni celle des formules, une institution s'intéresse à décrire les rapports existant entre la syntaxe et la sémantique, *i.e.* entre les ensembles de formules définis au-dessus des signatures et les catégories de modèles associées à ces dernières. C'est en quelque sorte une logique à laquelle manque tous les mécanismes d'inférence mais qui permet de capturer formellement la notion intuitive de système logique d'un point de vue théorie des modèles (les *logiques générales* de J. Meseguer (*cf.* [90]) sont une extension des institutions permettant de capturer ces mécanismes d'inférence). De ce fait, les institutions constituent une théorie abstraite des modèles utilisant la théorie des catégories pour langage. De plus, la notion de *flèche*, ou *morphisme*, entre institutions a permis de mettre en relation des systèmes logiques présentés par des institutions différentes, ce qui permet, d'une part de comparer ces systèmes entre eux (*cf.* [28]) et, d'autre part, de définir l'emprunt de fonctionnalités diverses telles que des systèmes d'inférence, des prouveurs de théorèmes, *etc.* (*cf.* [29]).
D'autres cadres théoriques ont été proposés à la suite des institutions comme par exemple les *langages de spécification* de B. Mahr et J.A. Makowsky (*cf.* [80, 79, 81]), les pré-institutions de A. Salibra et G. Scollo (*cf.* [112]) ou les différentes variantes des parchemins (*cf.* [62, 98, 26, 25]). Néanmoins, leurs applications se sont le plus souvent concentrées sur la résolution de problèmes bien particuliers (théorie des modèles dans le cadre de la logique mathématique pour les logiques générales de J. Barwise, affaiblissement du cadre des institutions par les pré-institutions, *etc.*) ou bien sont restées attachées à un certain groupe de travail (les c-parchemins et la combinaison de logiques) et n'ont pas eu la résonance des institutions.

L'approche institution-indépendante

Comme nous l'avons vu précédemment, en mathématiques, c'est le manque de pouvoir d'expression qui est à l'origine de la définition d'une nouvelle logique. De ce fait, les résultats qui ont été généralisés dans le cadre des logiques abstraites de J. Barwise sont principalement liés à ces considérations de pouvoir d'expression (voir le théorème de Lindström succinctement présenté ci-dessus). À l'inverse, dans le cadre du génie logiciel formel, parce que les spécifications

écrites vont être exploitées pour renforcer la correction des logiciels, c'est l'expressivité des formalismes qui nous intéresse et donc des caractéristiques telles que la lisibilité ou la clarté des spécifications, ou encore la façon de les structurer, de faire du prototypage, de mener des preuves, *etc.* Dans le cadre des institutions de nombreux travaux ont alors été effectués pour formaliser ces sujets propres à la conception de logiciels. Parmi les plus étudiés, c'est surtout la structuration de spécifications (*cf.* [21, 104, 27]), la modularité (*cf.* [16, 45, 104]), le raffinement (*cf.* [21, 49, 117]) et le test fonctionnel (*cf.* [7]) qui ont reçus le plus d'attention à ce niveau d'abstraction. Dans le cas de la structuration, par exemple, T. Borzyszkowski a montré dans [21] comment définir un ensemble d'opérateurs permettant de structurer des spécifications et conserver la complétude du système de preuve lorsqu'on étend celui-ci aux spécifications structurées. Cette étude de la structuration a été faite directement dans le cadre des institutions, *i.e.* indépendamment de la logique sous-jacente. Pour bénéficier directement de ces résultats, il suffit alors de présenter la logique considérée par une institution et d'instancier les résultats de [21].
Maintenant, certaines de ces techniques de conception et de vérification sont fondées sur des résultats importants de la théorie standard des modèles. On peut par exemple citer les notions d'existence de quotients et de modèles libres, ou encore le fait de posséder des propriétés telles que l'amalgation ou l'interpolation de Craig qui sont sous-jacentes à la modularité et à la structuration. Il devient alors important de s'intéresser dans le cadre des institutions à ces résultats pour eux-mêmes, *i.e.* il est important de généraliser ces résultats indépendamment de la logique sous-jacente de façon à ce qu'ils soient utilisables pour le plus grand nombre de systèmes logiques. Or, les institutions ont initialement été introduites non seulement dans le but d'étudier des résultats de théorie de la spécification indépendamment de la logique sous-jacente, mais aussi pour la généralisation et l'étude de résultats issus de la théorie standard des modèles. Si, comme nous l'avons vu plus haut, il existe de nombreux travaux institution-indépendants en théorie de la spécification, peu ont pour sujet des problèmes de théorie des modèles. La plupart de ces travaux concernent d'ailleurs des résultats sous-jacents à des problèmes de théorie de la spécification. Ainsi en est-il des travaux de A. Tarlecki sur les notions de (quasi)-variétés et de modèles initiaux (généralisation des théorèmes de Birkhoff, McKinsley et Mal'cev *cf.* [129]) et de ceux de A. Salibra et G. Scollo qui remontent au niveau des institutions l'équivalence (établie dans le cadre de la logique du premier ordre classique) de l'interpolation de Craig et de la consistance de Robinson, ainsi que la notion de compacité et les théorèmes de Löwenheim-Skolem (*cf.* [113, 114]). Depuis quelques années, R. Diaconescu s'intéresse à la généralisation dans le cadre des institutions de différents résultats de la théorie standard des modèles, indépendamment de leur intérêt en théorie de la spécification (*cf.* [40, 41, 42, 39, 38]). Cependant, ces travaux sont pour la plupart très récents et il reste encore beaucoup de résultats de théorie standard des modèles à généraliser dans le cadre des institutions. Ceci est d'autant plus important que certains de ces résultats constituent des conditions suffisantes pour des résultats de théorie de la spécification (par exemple l'interpolation de Craig pour la modularité).
Dans le cadre de cette thèse nous étudierons, au sein de la théorie des institutions, les résultats suivants issus de la théorie standard des modèles :

1. Méthode des diagrammes et morphismes élémentaires ;

2. Union des chaines de Tarski ;
3. Consistance de Robinson ;
4. Interpolation de Craig
5. Théorèmes de Löwenheim-Skolem ;
6. Théorème de Beth.

Combinaison de logiques

Comme nous l'avons exposé plus haut, le fait de travailler indépendamment du système logique sous-jacent en présentant celui-ci par une institution a permis de nombreuses avancées en théorie de la spécification (théorie du test, structuration, raffinement, prototypage, *etc.*). Cependant, travailler dans un système logique arbitraire mais fixe n'est pas toujours suffisant. Il existe deux raisons à cela. D'une part, les systèmes à spécifier sont toujours plus complexes et englobent toujours plus de fonctionnalités. Afin de décrire au mieux de tels systèmes, il est alors nécessaire de faire appel à différents paradigmes de spécification, *i.e.* utiliser conjointement plusieurs systèmes logiques. D'autre part, la réutilisation des spécifications est une donnée importante en génie logiciel car elle ouvre la voie à la création de bibliothèques de spécifications et à la spécification de systèmes par assemblage de composants « pris sur l'étagère » et donc éventuellement décrits par des formalismes différents.
Considérant la profusion actuelle de formalismes et de systèmes logiques, il est indispensable de gérer la diversité existante de systèmes logiques, *i.e.* il faut s'intéresser aux relations entre systèmes logiques. Deux problèmes sont à étudier tout particulièrement car ils sont fortement liés à la spécification de systèmes complexes :

1. *la représentation d'un système logique dans un autre* en vue de réutiliser des outils tels que des prouveurs de théorèmes ;
2. *la combinaison de systèmes logiques* en vue d'introduire de façon modulaire de nouveaux concepts à un système logique donné.

La première de ces deux opérations est étudiée depuis longtemps et a fait l'objet de nombreux travaux. Elle se traduit par différentes notions de flèches entre institutions, telles les *morphismes d'institutions*, les *transformations de (pré)-institutions*, les *simulations*, *etc.*, chacune exprimant une notion différente de représentation de systèmes logiques (*cf.* [86, 28, 132]). La définition de chacune de ces flèches vient en général avec un ensemble de conditions permettant, suivant les cas, la préservation de résultats ou le transfert d'outils.
Différentes méthodes de combinaison de logiques ont été étudiées telles que la synchronisation, la paramétrisation ou encore la fibration. Malgré certains travaux dans le contexte général des institutions (*cf.* [61, 62, 52, 90, 63, 73, 124, 94, 132, 8, 29, 97, 98, 106]), la plupart des travaux sur ce sujet ont été réalisés dans le cadre des logiques modales (*cf.* [135, 72, 53, 54, 143, 56, 57, 59, 87, 74, 88, 58, 144]), la plupart du temps avec différentes motivations. Cependant, de récents travaux (*cf.* [97, 98, 25, 26, 123, 122]) ont étendu la notion originelle des parchemins (*cf.* [62]) justement dans le but de fournir le degré d'abstraction nécessaire pour définir des mécanismes sophistiqués de combinaison de logiques. De nombreux travaux ont alors été réalisés dans ce cadre, principalement par le groupe

de recherche FibLog dirigé par A. Sernadas (*cf.* [24, 25, 125, 144, 23, 32, 31].

La littérature sur le sujet montre que la combinaison de logiques n'a été étudiée que sous l'angle de la combinaison (en fait une simple union des connecteurs logiques et des quantificateurs) de deux logiques connues et possédant des propriétés intéressantes. Ceci détermine alors de manière unique une nouvelle logique qui, éventuellement, possède les propriétés des logiques sources. Or, ceci ne prend en compte qu'une partie du problème de la combinaison de logiques. En effet, dans la pratique, l'opération inverse (*i.e.* voir une logique « définie *ex-nihilo* » comme la combinaison de logiques existantes) est tout autant indispensable, si ce n'est plus. Intuitivement, il s'agit de voir une logique donnée comme une combinaison de logiques connues, moyennant certaines modifications éventuelles. L'intérêt de considérer la combinaison de logiques sous cet angle est, entre autres, de mieux prendre en compte l'extension de formalismes à un nouvel aspect et permettre ainsi le transport de propriétés vers cette nouvelle logique. C'est donc sous cet angle que nous étudierons la combinaison de logiques dans cette thèse, *i.e.* nous définirons la combinaison de logiques comme une opération en deux étapes : une modification de logiques existantes (au travers d'une notion particulière de flèche entre logiques) et un mélange de ces modifications (par l'union des ensembles de constructeurs de formules). Notons que cette définition de la combinaison de logiques généralise celles données par de précédents travaux, ces dernières réduisant les modifications des logiques sources à l'identité.

Bien entendu, que ce soit dans les applications ou au niveau théorique, la combinaison de logiques n'a d'intérêt que dans l'optique de transfert de propriétés et d'outils d'une (ou plusieurs) logique(s) à une autre. C'est d'ailleurs le cas de la plupart des travaux existant sur la combinaison de logiques qui se sont intéressés à la préservation de propriétés telles que la correction, la complétude ou encore la décidabilité. Dans cette thèse, nous nous attacherons à l'étude de la préservation de propriétés plus en rapport avec les besoins des spécifieurs telles que, par exemple, celles sous-jacentes à la structuration et la modularité des spécifications. L'interpolation de Craig et la consistance de Robinson étant des propriétés cruciales pour ces dernières, il est naturelle de s'intéresser en priorité à la préservation de ces propriétés au travers de la combinaison de logiques.

Plan de thèse

Les travaux que nous présentons ici portent sur la structuration des spécifications formelles de type axiomatique et la préservation de la structuration au travers de la combinaison de logiques. La présentation de ce travail est scindée est trois parties distinctes, chacune correspondant à un aspect particulier.

La première partie concerne l'étude de l'*interpolation de Craig* et de son équivalence avec la *consistance de Robinson* dans le cadre des institutions. Nous présentons dans un premier temps les institutions et rappelons un certain nombre de résultats qui nous seront utiles pour la suite. Le deuxième chapitre de cette partie est consacré à l'étude de l'interpolation de Craig. Après un bref historique de l'étude de cette propriété et un exposé des motivations de cette étude, nous présentons les approches différentes suivies indépendamment par R. Diaconescu (*cf.* [41]) et A. Salibra et G. Scollo (*cf.* [114]) établissant dans un cas l'interpolation de Craig pour une classe de logiques dont les catégories de modèles possèdent certaines propriétés de fermeture et, dans l'autre cas généralisant l'équivalence entre la propriété d'interpolation de Craig et le lemme de consistance de Robinson en présence de la négation. Nous clôturons cette première partie par un chapitre consacré à la généralisation dans le cadre des institutions du théorème de définissabilité de Beth qui est, dans le cadre de la logique du premier ordre classique, une conséquence de l'interpolation de Craig. Ce résultat n'est pas directement lié à la théorie de la spécification. Il est en fait assez anecdotique au regard du but premier de cette thèse qui est la généralisation des résultats de théorie standard des modèles sous-jacents à certaines techniques d'aide au développement (structuration, modularité). De ce fait, il peut donc être laissé de côté en première lecture, d'autant plus que, contrairement aux deux premiers chapitres, il ne fait pas partie de l'état de l'art de cette thèse[6].

L'état de l'art sur l'interpolation de Craig donné dans cette première partie met en évidence l'importance de cette propriété pour la théorie de la spécification et donc la nécessité de donner des conditions pour que celle-ci soit vérifiée. L'équivalence de l'interpolation de Craig et du lemme de consistance de Robinson nous fournit une telle condition. Cependant, la propriété de Robinson, dont le lemme de consistance est une conséquence directe, repose sur la notion d'équivalence élémentaire de modèles. Or, cette dernière propriété est difficile à montrer dans la pratique. En théorie standard des modèles, il existe une méthode permettant de construire des modèles élémentairement équivalents à un modèle donné. Cette méthode s'appelle la *méthode des diagrammes* et repose sur la notion de satisfaction des formules et donc, sur les éléments permettant cette satisfaction. Or, la théorie des institutions ne prend pas en compte cette notion d'éléments servant à la satisfaction des formules et n'indique pas comment est définie la relation de satisfaction des formules dans les modèles. Dans cette deuxième partie nous proposons donc une extension des institutions permettant de prendre en compte les éléments servant à la satisfaction des formules et que nous appelons *institutions stratifiées*. Nous définissons alors dans ce nouveau cadre théorique la méthode des diagrammes ce qui nous permet ensuite de fournir des conditions suffisantes pour obtenir le lemme de consistance de Robinson, et donc, sous réserve de posséder la négation, pour l'interpolation de Craig.

L'étude de la consistance de Robinson dans le cadre des institutions stratifiées montre que le théorème des chaînes élémentaires de Tarski en est une des conditions suffisantes. Afin de démontrer ce théorème, nous avons été amenés à étendre un peu plus notre cadre théorique pour rendre explicite la notion de constructeurs de formules. Ceci étant fait, la structure des formules devient elle aussi explicite, les formules étant alors des termes sur une algèbre particulière. De ce fait, il était alors naturel de s'intéresser au problème de la combinaison de logiques. Stipulant qu'une combinaison de logiques ne se résume pas à une simple union d'ensembles de constructeurs de formules, nous proposons une dé-

6. À notre connaissance, il n'existe pas de généralisation du théorème de Beth dans le cadre des institutions (ni même dans un cadre logique-indépendant).

finition de la combinaison de logiques consistant en deux étapes. Tout d'abord une modification de logiques existantes, dites « sources », représentée par un morphisme puis, une union des logiques sources modifiées représentée par une partition de l'ensemble des constructeurs de formules de la logique combinée. Bien entendu, la combinaison de logiques n'a d'intérêt que dans la perspective d'un transfert de résultats des logiques sources vers la logique combinée. Nous nous sommes alors naturellement intéressé à la préservation de l'interpolation de Craig et du lemme de consistance de Robinson au travers de cette combinaison.

Ce travail étant très théorique, il peut être perçu comme relevant plus des mathématiques que de l'informatique à proprement parler. Nous soutenons cependant que, dans l'optique actuelle de sécurité des systèmes informatiques, un tel travail est indispensable. Prenant acte de l'importance de la structuration et du raffinement des spécifications formelles dans le processus de développement de logiciels (*cf.* [67, 115, 116, 117, 118, 119, 120, 121, 134, 21, 27, 141, 142]), ainsi que des besoins exprimés par le monde industriel, notamment en terme de modularité et de réutilisation des spécifications formelles (spécifications hétérogènes, composants sur l'étagère, réutilisation de preuves, d'outils, *etc.*), notre travail s'inscrit dans une continuité de travaux portant sur les fondements de l'informatique et, en dépit de son caractère très théorique, trouve sa justification dans le fait qu'il est indispensable de fournir des bases solides à l'utilisation des spécifications formelles dans le cadre de l'aide au développement de logiciels de qualité. Sans de tels travaux, les méthodes formelles perdraient ce qui les distingue des méthodes non-formelles et fait leur valeur, à savoir la rigueur et la non-ambiguïté des spécifications.

Nous supposons dans tout le document un certain degré de familiarité avec les notations usuelles de la théorie des ensembles et de la théorie des catégories. Les notations ensemblistes utilisées dans ce document seront définies au moment de leur première utilisation, laquelle figure dans l'index des notations situé en fin de document. En ce qui concerne les notions catégorielles utilisées, nous renvoyons le lecteur à l'annexe A.

Première partie

Interpolation de Craig dans les institutions

Les institutions ont été initialement introduite dans le but de généraliser des résultats aussi bien en théorie de la spécification qu'en théorie des modèles, indépendamment de la logique sous-jacente. Cependant, ce sont essentiellement des résultats de théorie de la spécification qui ont été généralisés à l'aide des institutions. En particulier, le cadre offert par les institutions a permis de définir independamment du système logique sous-jacent des techniques de développement telles que le test (*cf.* [77]), la structuration (*cf.* [21, 104]), le raffinement (*cf.* [21, 116]), le prototypage (*cf.* [9]), *etc.*
Compte tenu des bénéfices importants apportés à la théorie de la spécification par la théorie des institutions, il est déjà traditionnel dans la pratique académique de représenter chaque formalisme utilisé pour la spécification d'un système donné par une institution. Ceci permet alors de profiter de résultats généraux établis dans le cadre des institutions tels que la définition d'un ensemble adéquate d'opérateurs de structuration des spécifications, la définition d'un système d'inférence correcte par rapport à la sémantique ou la conservation de cette propriété lorsqu'on étend le système d'inférence aux spécifications structurées (*cf.* [21]). D'ailleurs, la plupart des langages récents de spécifications algébriques suivent cette tradition comme par exemple CASL (*cf.* [95]), Maude (*cf.* [91]) ou encore CafeOBJ (*cf.* [44]).
Cependant, et bien que cette vue orientée modèle des logiques soit un outil théorique puissant pour l'étude des systèmes logiques et de leur sémantique, peu de chercheurs se sont intéressés à la généralisation de résultats issus de la théorie standard des modèles dans le cadre des institutions. Il existe tout de même quelques travaux isolés sur le sujet tel que :

- les travaux de A. Tarlecki ([129, 131]) et sa notion d'institutions algébriques abstraites, cadre dans lequel l'auteur généralise des résultats algébriques classiques tels que le théorème de Birkhoff (équivalence entre les théories équationnelles et les variétés), le théorème de McKinsley (équivalence entre les théories universelles de Horn et les quasi-variétiés) et le théorème de Mal'cev (existence de modèles initiaux de termes pour les théories universelles de Horn) ;
- les travaux de A. Salibra et G. Scollo ([113, 114]) concernant les relations entre les propriétés d'interpolation, de compacité et de Löweinem-Skolem dans la categorie des pré-institutions [7] (*cf.* [112]).
- les travaux de R. Diaconescu ([38, 39, 42, 41]) proposant une preuve de l'interpolation de Craig indépendamment de l'institution considérée ([42, 41]) ainsi qu'une étude, dans le cadre des institutions, de différentes propriétés issues de la théorie des modèles telles que les ultraproduits ([39]).

Comme on le voit, les recherches en théorie des modèles dans le cadre des institutions sont relativement marginales alors même que la théorie des modèles traite de problèmes touchant aux fondements de l'informatique et plus particulèrement de la théorie de la spécification. Il a par exemple été montré que l'interpolation de Craig est directement liée à certains aspects de la structuration et de la modularité des spécifications (*cf.* [21, 47, 138, 139]) ainsi qu'à certains problèmes de complexité (*cf.* [100, 101]). Cette situation est alors en contraste avec le sentiment partagé par certains chercheurs sur le fait que les concepts et résultats de théorie standard des modèles, largement dépendants

7. Les pré-institutions sont un affaiblissement des institutions essentiellement dans le sens où elles n'imposent pas la condition de satisfaction.

du cadre de la logique du premier ordre et de ses particularités, peuvent être clarifiés et reformulés plus conformément à l'intuition grâce à la théorie des institutions.

Dans cette première partie, nous nous intéressons alors à l'*interpolation de Craig dans le cadre des institutions.* L'interpolation de Craig est une propriété de théorie standard des modèles que W. Craig a montrée pour la logique du 1^{er}ordre avec égalité (*cf.* [34]). Du fait des liens très forts existant entre cette propriété et la notion de modularité des systèmes informatiques, de nombreux travaux ont cherché à généraliser cette propriété, que ce soit dans le cadre de la logique du 1^{er}ordre (*cf.* [13]), dans un cadre institution indépendant (*cf.* [45, 47]), dans le cadre de spécifications structurées (*cf.* [27, 21]) ou encore dans le cadre de spécifications hétérogènes (*cf.* [133]). Ces différents travaux ont permis de trouver une formulation de l'interpolation de Craig plus adaptée à la théorie de la spécification que la formulation originelle de cette propriété pour la logique du premier ordre classique. En particulier, cette formulation s'appuie sur la notion de morphisme de signatures indispensable en théorie de la spécification et fait référence à des ensembles de formules plutôt qu'à des formules simples[8]. Dans cette première partie, nous nous intéressons à l'interpolation de Craig essentiellement d'un point de vue théorie des modèles et nous étudions notamment les relations d'interdépendance entre la propriété d'interpolation de Craig et un résultat fondamental de théorie standard des modèles dû à Robinson (*cf.* [109]) : le lemme de consistance.
À la fin de cette première partie, nous portons notre attention sur le théorème de définissabilité de Beth. Ce théorème, montré par E. Beth pour la logique du 1^{er}ordre classique (*cf.* [15]), est une des conséquences les plus importantes de l'interpolation de Craig. Il établit le fait que tout prédicat implicitement définissable[9] est explicitement définissable[10]. La notion de morphisme de signatures n'étant que de peu d'intérêt pour le logicien, les propriétés de définisabilité implicite et explicite ont principalement été étudiées dans le cas d'inclusions de signatures. Dans le cadre de la théorie standard des modèles, le théorème de Beth n'est alors valable que pour des inclusions de signatures. Or, parce qu'elle dénote des relations diverses entre les signatures telles que des changements de langage, la notion de morphisme de signatures est cruciale en théorie de la spécification. Intuitivement, elle est également liée à la notion de définissabilité. Nous avons donc étendu les deux définitions de définissabilité à tout morphisme de signatures et reformulé le théorème de Beth en conséquence.

Le plan de cette partie est le suivant. Le premier chapitre est un chapitre introductif. On y rappelle la définition des institutions ainsi que l'ensemble des résultats et définitions liés à la théorie de la spécification qui nous seront utiles par la suite. Le lecteur familier avec les institutions et la théorie de la spécification peut aisément passer ce chapitre. Le deuxième chapitre présente, après

8. Notons, en particulier, que la logique équationnelle vérifie la version « ensemble de formules » de l'interpolation de Craig mais pas sa version « formule » (*cf.* [110]) qui requiert nécessairement la conjonction

9. Un prédicat est implicitement définissable si sa définition est en termes de modèles (*cf.* définition 3.2.1)

10. Un prédicat est explicitement définissable si sa définition est en termes de formules (*cf.* définition 3.2.2)

un cours état de l'art sur l'interpolation de Craig, deux approches différentes pour démontrer cette propriété dans un cadre institution-indépendant. La première, proposée par R. Diaconescu, s'appuie sur la notion d'axiomatisabilité tandis que la deuxième, proposée par A. Salibra et G. Scollo, fait appel à la notion de consistance définie par Robinson. Nous proposons dans ce chapitre une légère amélioration de cette dernière approche en affaiblissant certaines conditions suffisantes et étudions les conditions de transfert de l'interpolation de Craig au travers de morphismes d'institutions. Nous clôturons cette première partie par un chapitre consacré au théorème de définissabilité de Beth qui est une conséquence directe du théorème d'interpolation de Craig dans le cadre de la logique du premier ordre classique. Après en avoir donné une preuve institution-indépendante, nous étudions les conditions de transfert au travers des morphismes d'institutions et nous généralisons la notion de définissabilité à tout morphisme de signature.

Chapitre 1

Institutions

Nous présentons dans ce chapitre les différents éléments de la théorie abstraite de la spécification (*i.e.* logique-indépendante) sur lesquels porte notre travail. Le cadre dans lequel nous nous plaçons est celui des institutions. Ces dernières offrent un cadre théorique permettant de capturer de façon formelle la notion de système logique d'un point de vue théorie des modèles. Nous en rappelons ici la définition et illustrons notre propos par des exemples d'institutions bien connues. Nous exposons ensuite les récents travaux de T. Mossakowski, J. Goguen, R. Diaconescu et A. Tarlecki (*cf.* [93]) sur la représentation implicite des connecteurs logiques et des quantificateurs dans les institutions. L'un des buts premiers de la théorie des institutions étant l'étude des spécifications et de leur sémantique, nous introduisons en section 1.3 les notions de présentation et de théorie, qui formalisent le concept de spécification à plat, et les propriétés usuelles qui leur sont associées. En section 1.4, nous présentons certaines propriétés sémantiques particulièrement importantes pour l'étude des spécifications et de leur sémantique. Enfin, en section 1.5, nous reprenons les travaux de T. Borzyszkowski concernant la structuration des spécifications axiomatiques (*cf.* [21]). Les institutions ne s'occupant que de la partie théorie des modèles des systèmes logiques qu'elles présentent, différentes extensions, telles que les π-institutions de J. Fiadeiro (*cf.* [52]) et les logiques générales de P. Meseguer (*cf.* [90]), ont été proposées de façon à prendre en compte la partie théorie de la preuve des systèmes logiques. Nous introduisons alors la notion d'inférence en section 1.6 et montrons, toujours d'après les travaux de T. Borzyszkowski, comment étendre les systèmes d'inférence aux spécifications structurées. Nous proposons enfin un court exposé des principes essentiels de la modularité et clôturons ce chapitre par la présentation des deux notions principales de flèches entre institutions : les morphismes et les co-morphismes.
Ce chapitre est essentiellement pour nous l'occasion de fixer les notations que nous utiliserons par la suite et de définir un certain nombre de propriétés sémantiques importantes sur lesquelles porte notre travail. Le lecteur familier avec ces différents concepts et désireux d'aborder sans attendre les apports de cette thèse à la théorie de la spécification pourra passer directement au chapitre 2.

1.1 Définition

La théorie des institutions (*cf.* [63]) a été introduite au début des années 80 par J. Goguen et R. Burstall à la suite de leurs travaux sur la sémantique du langage Clear (*cf.* [22]). Cherchant à « effectuer tout le travail de Clear une fois pour toute, indépendemment du système logique sous-jacent » [1], ils ont réussi à caractériser de façon générique les concepts indispensables pour qu'une théorie fournisse un cadre adapté à la conception de systèmes informatiques. Une institution formalise alors le concept de système logique en donnant :

- une catégorie de *signatures*. Ces dernières fournissent le vocabulaire nécessaire à la description d'un problème donné. Intuitivement, les signatures sont aux systèmes logiques ce que les interfaces sont aux programmes. Les *morphismes de signatures*, qui dénotent des relations entre signatures, sont à rapprocher de la notion de modification d'interfaces (enrichissement, surcharge, *etc.*) ;
- à chaque signature Σ est associé un ensemble de formules closes [2], appelées Σ-*formules*, auquel on étend les morphismes de signatures ;
- à chaque signature Σ est associée une catégorie de *modèles*, appelés Σ-*modèles*, et on étend les morphismes de signatures aux catégories de modèles ;
- à chaque signature Σ est associée une *relation de satisfaction* entre les Σ-formules et les Σ-modèles.

De plus, on impose le fait que le changement de signatures selon un morphisme de signatures n'affecte pas la satisfaction des formules par les modèles. Par analogie avec les langages de programmation, ceci traduit le fait qu'un programme peut faire quelque chose de plus simple que ce pourquoi il a été conçu (par exemple un programme permettant de manipuler des piles, et en particulier de calculer la cardinalité de ces dernières, est capable de faire de l'arithmétique élémentaire sur les entiers).

Ainsi, une institution distingue les signatures des formules au sein des théories et les modèles sémantiques associés aux signatures permettent de donner une valeur de vérité aux formules. Sans pour autant détailler la structure des modèles ni celle des formules, une institution s'intéresse à décrire les rapports existant entre la syntaxe et la sémantique, *i.e.* entre les ensembles de formules définis au-dessus des signatures et les catégories de modèles associées à ces dernières. C'est en quelque sorte une logique à laquelle manquent tous les mécanismes d'inférence mais qui permet de capturer formellement la notion vague de système logique d'un point de vue théorie des modèles.

Définition 1.1.1 (institution) *Une* institution *est la donnée d'un quadruplet* $(\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ *tel que* [3] *:*

- $\mathcal{Sig}$ *est une* catégorie *dont les objets sont appelés des* signatures *;*

1. *In* [62].
2. Une formule close est une formule dont toutes les variables sont sous la portée d'un quantificateur, *i.e.* sans variable libre.
3. De façon à éviter tout problème fondationnel relatif à la catégorie des catégories, nous considérons ici que $\mathcal{Cat}$ désigne la catégorie de toutes les catégories dont les objets et les morphismes appartiennent à un univers donné (*cf.* annexe A) dont on ne fera pas mention (*cf.* [63, 90, 116] pour des remarques similaires).

- $\mathcal{S}en : \mathcal{S}ig \to \mathcal{S}et$ *est un* foncteur *qui associe à toute signature* $\Sigma \in |\mathcal{S}ig|$ *l'ensemble* $\mathcal{S}en(\Sigma) \in |\mathcal{S}et|$ *des formules (closes) construites au-dessus de* Σ. *On appelle* Σ-formules *les éléments de* $\mathcal{S}en(\Sigma)$ *;*
- $\mathcal{M}od : \mathcal{S}ig \to \mathcal{C}at$ *est un* foncteur contravariant *qui associe à toute signature* $\Sigma \in |\mathcal{S}ig|$ *la catégorie* $\mathcal{M}od(\Sigma) \in |\mathcal{C}at|$ *des modèles de* Σ. *On appelle* Σ-modèles *les objets de* $\mathcal{M}od(\Sigma)$ *;*
- $\models\ = (\models_\Sigma)_{\Sigma \in |\mathcal{S}ig|}$ *est une* famille de relations binaires *telle que pour toute signature* $\Sigma \in |\mathcal{S}ig|$, $\models_\Sigma \subseteq |\mathcal{M}od(\Sigma)| \times \mathcal{S}en(\Sigma)$ *est la* relation de satisfaction *des* Σ*-formules par les* Σ*-modèles.*

De plus, pour tout morphisme de signature $\sigma : \Sigma \to \Sigma'$, *tout* Σ'*-modèle* $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ *et toute* Σ*-formule* $\varphi \in \mathcal{S}en(\Sigma)$, *on a :*

$$\mathcal{M}' \models_{\Sigma'} \mathcal{S}en(\sigma)(\varphi) \Leftrightarrow \mathcal{M}od(\sigma)(\mathcal{M}') \models_\Sigma \varphi \tag{1.1}$$

La formule (1.1) ci-dessus est appelée *condition de satisfaction.* Le sens ($\Rightarrow$) est appelé *préservation de la satisfaction par réduction (psr)* et établit le fait que la satisfaction des Σ-formules par les Σ'-modèles sur Σ' est conservée lorsqu'on se place dans Σ, *i.e.* lorsqu'on considère la réduction des Σ'-modèles aux symboles de Σ. Le sens ($\Leftarrow$) est appelé *préservation de la satisfaction par expansion (pse)* et établit le fait que la satisfaction sur Σ des Σ-formules par les réduits des Σ'-modèles aux symboles de Σ est conservée lorsqu'on se place dans Σ'.

Notation 1.1.1 *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution et soit* $\Sigma \in |\mathcal{S}ig|$ *une signature quelconque. Soient* $\varphi \in \mathcal{S}en(\Sigma)$ *et* $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$. *On notera* $\mathcal{M} \not\models_\Sigma \varphi$ *si* $\mathcal{M} \models_\Sigma \varphi$ *est faux, i.e. si* $\mathcal{M}$ *ne satisfait pas* φ.

Nous présentons maintenant quelques exemples classiques d'institutions.

Exemple 1.1.1 (logique propositionnelle) *L'institution* PL *de la logique propositionnelle est définie de la façon suivante :*

- *la catégorie* $\mathcal{S}ig_{\mathrm{PL}}$ *des* signatures propositionnelles *est la catégorie* $\mathcal{S}et$ *des ensembles. Un élément d'une signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{PL}}|$ *est appelé* variable propositionnelle *;*
- $\mathcal{S}en_{\mathrm{PL}} : \mathcal{S}ig_{\mathrm{PL}} \to \mathcal{S}et$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{PL}}|$ *l'ensemble* $\mathcal{S}en_{\mathrm{PL}}(\Sigma) \in |\mathcal{S}et|$ *des* Σ-formules propositionnelles, *i.e. le plus petit ensemble au sens de l'inclusion contenant la signature* Σ *et fermé par la négation* ($\neg$) *et la disjonction* ($\vee$), *et répercute les changements de signatures sur les formules ;*
- $\mathcal{M}od_{\mathrm{PL}} : \mathcal{S}ig_{\mathrm{PL}} \to \mathcal{C}at$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{PL}}|$ *la catégorie* $\mathcal{M}od_{\mathrm{PL}}(\Sigma) \in |\mathcal{C}at|$ *des* Σ-valuations, *i.e. la catégorie des applications* $v : \Sigma \to \{0, 1\}$ *qui associe une valeur de vérité à toute variable propositionnelle*[4]. *Étant donné un morphisme de signatures* $\sigma \in \mathcal{S}ig_{\mathrm{PL}}(\Sigma, \Sigma')$, *le réduit* $\mathcal{M}od_{\mathrm{PL}}(\sigma)(v')$ *d'un* Σ'*-modèle* $v' \in |\mathcal{M}od_{\mathrm{PL}}(\Sigma')|$ *est la* Σ*-valuation* $v \in |\mathcal{M}od_{\mathrm{PL}}(\Sigma)|$ *obtenue par composition du morphisme* σ *avec la* Σ'*-valuation* v', *i.e.* $v = v' \circ \sigma$ *;*

4. Dans la littérature, et en particulier dans les travaux en rapport avec l'informatique, les booléens sont couramment représentés par 0 (*faux* [F]) et 1 (*vrai* [V]). Les traductions anglaises de « faux » (*false* [F]) et « vrai » (*true* [T]) sont également très usitées.

- $\models_{\mathrm{PL}} = (\models^{\mathrm{PL}}_{\Sigma})_{\Sigma \in |\mathcal{Sig}_{\mathrm{PL}}|}$ *est une famille de relations binaires telle que pour toute signature* $\Sigma \in |\mathcal{Sig}_{\mathrm{PL}}|$*, la relation* $\models^{\mathrm{PL}}_{\Sigma} \subseteq |\mathcal{Mod}_{\mathrm{PL}}| \times \mathcal{Sen}_{\mathrm{PL}}(\Sigma)$ *est définie de la façon suivante (où* $v \in |\mathcal{Mod}_{\mathrm{PL}}(\Sigma)|$*) :*
  - $\forall p \in \Sigma,\ v \models^{\mathrm{PL}}_{\Sigma} p$ *ssi* $v(p) = 1$,
  - $\forall \varphi \in \mathcal{Sen}_{\mathrm{PL}}(\Sigma),\ v \models^{\mathrm{PL}}_{\Sigma} \neg\varphi$ *ssi* $v \not\models^{\mathrm{PL}}_{\Sigma} \varphi$,
  - $\forall \varphi, \psi \in \mathcal{Sen}_{\mathrm{PL}}(\Sigma),\ v \models^{\mathrm{PL}}_{\Sigma} \varphi \vee \psi$ *ssi* $v \models^{\mathrm{PL}}_{\Sigma} \varphi$ *ou* $v \models^{\mathrm{PL}}_{\Sigma} \psi$.

Proposition 1.1.1 *Pour tout morphisme de signatures* $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$*, tout* Σ'*-modèle* $v' \in |\mathcal{Mod}_{\mathrm{PL}}(\Sigma')|$ *et toute* Σ*-formule* $\varphi \in \mathcal{Sen}_{\mathrm{PL}}(\Sigma)$*,* $v' \models^{\mathrm{PL}}_{\Sigma'} \mathcal{Sen}_{\mathrm{PL}}(\sigma)(\varphi)$ *si et seulement si* $\mathcal{Mod}_{\mathrm{PL}}(\sigma)(v') \models^{\mathrm{PL}}_{\Sigma} \varphi$.

Preuve La preuve se fait par induction sur la structure des formules.
Notons tout d'abord que pour toute variable propositionnelle $p \in \Sigma$, on a $v(p) = v'(\sigma(p))$ ce qui prouve la condition de satisfaction pour les variables propositionnelles. La préservation de la condition de satisfaction par les connecteurs logiques se vérifie simplement d'après les définitions. □

Exemple 1.1.2 (logique propositionnelle modale) *L'institution* PML *de la logique propositionnelle modale est définie de la façon suivante :*

- *la catégorie* $\mathcal{Sig}_{\mathrm{PML}}$ *est la catégorie* $\mathcal{Sig}_{\mathrm{PL}}$ *des signatures propositionnelles ;*
- $\mathcal{Sen}_{\mathrm{PML}} : \mathcal{Sig}_{\mathrm{PML}} \to \mathcal{Set}$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{Sig}_{\mathrm{PML}}|$ *l'ensemble* $\mathcal{Sen}_{\mathrm{PML}}(\Sigma) \in |\mathcal{Set}|$ *des* Σ-formules propositionnelles modales, *i.e. le plus petit ensemble au sens de l'inclusion contenant la signature* Σ *et fermé par la négation (*$\neg$*), la disjonction (*$\vee$*) et la modalité* $\Box$*, et répercute les changements de signatures sur les formules ;*
- $\mathcal{Mod}_{\mathrm{PML}} : \mathcal{Sig}_{\mathrm{PML}} \to \mathcal{Cat}$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{Sig}_{\mathrm{PML}}|$ *la catégorie* $\mathcal{Mod}_{\mathrm{PML}}(\Sigma) \in |\mathcal{Cat}|$ *des* Σ-modèles de Kripke, *i.e. la catégorie dont les objets sont des triplets* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$ *où* $\mathfrak{E}$ *est un ensemble non vide d'états,* $\mathfrak{R} \subseteq \mathfrak{E} \times \mathfrak{E}$ *est une relation binaire sur* $\mathfrak{E}$ *appelée* relation d'accessibilité *et* $\mathfrak{v} : \mathfrak{E} \times \Sigma \to \{0, 1\}$ *est une application qui à tout couple* `(état,variable propositionnelle)` *associe une valeur de vérité. Étant donné un morphisme de signatures* $\sigma \in \mathcal{Sig}_{\mathrm{PML}}(\Sigma, \Sigma')$*, le réduit* $\mathcal{Mod}_{\mathrm{PML}}(\sigma)((\mathfrak{E}', \mathfrak{R}', \mathfrak{v}'))$ *d'un* Σ'*-modèle* $(\mathfrak{E}', \mathfrak{R}', \mathfrak{v}') \in |\mathcal{Mod}_{\mathrm{PML}}(\Sigma')|$ *est le* Σ*-modèle de Kripke* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \in |\mathcal{Mod}_{\mathrm{PML}}(\Sigma)|$ *tel que :*
  - $\mathfrak{E} = \mathfrak{E}'$ *;*
  - $\mathfrak{R} = \mathfrak{R}'$ *et la relation d'accessibilité est préservée, i.e. :*
    $\forall (\eta'_1, \eta'_2) \in \mathfrak{E}' \times \mathfrak{E}'$,
    $$\eta'_1 \mathfrak{R}' \eta'_2 \Rightarrow \eta'_1 \mathfrak{R} \eta'_2$$
  - *pour toute variable propositionnelle* $p \in \Sigma$ *et tout état* $\eta \in \mathcal{E}$*, on a :*
    $$\mathfrak{v}(p, \eta) = \mathfrak{v}'(\sigma(p), \eta)$$
- *la relation de satisfaction entre les modèles et les formules d'une même signature est stratifiée, pour chaque modèle, par les états associés à ce modèle. De ce fait,* $\models_{\mathrm{PML}} = (\models^{\mathrm{PML}}_{\Sigma})_{\Sigma \in |\mathcal{Sig}_{\mathrm{PML}}|}$ *est une famille de relations binaires telle que pour toute signature* $\Sigma \in |\mathcal{Sig}_{\mathrm{PML}}|$*, la relation* $\models^{\mathrm{PML}}_{\Sigma} \subseteq |\mathcal{Mod}_{\mathrm{PML}}| \times \mathcal{Sen}_{\mathrm{PML}}(\Sigma)$ *est définie de la façon suivante (où* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \in |\mathcal{Mod}_{\mathrm{PML}}(\Sigma)|$ *et* $\eta \in \mathfrak{E}$*) :*
  - $\forall p \in \Sigma,\ (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} p$ *ssi* $\mathfrak{v}(\eta, p) = 1$,

- $\forall \varphi \in \mathcal{S}en_{\mathrm{PML}}(\Sigma)$, $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \neg\varphi$ *ssi* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \not\models^{\mathrm{PML}}_{\Sigma,\eta} \varphi$,
- $\forall \varphi \in \mathcal{S}en_{\mathrm{PML}}(\Sigma)$, $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \Box\varphi$ *ssi pour tout* $\eta' \in \mathfrak{E}$ *tel que* $\eta\mathfrak{R}\eta'$, *on a* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta'} \varphi$,
- $\forall \varphi, \psi \in \mathcal{S}en_{\mathrm{PML}}(\Sigma)$, $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \varphi \vee \psi$ *ssi* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \varphi$ *ou* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \psi$.

Alors, $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma} \varphi$ *si pour tout* $\eta \in \mathfrak{E}$ *on a* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_{\Sigma,\eta} \varphi$.

Proposition 1.1.2 *Pour tout morphisme de signatures* $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$, *tout* Σ'*-modèle* $(\mathfrak{E}', \mathfrak{R}', \mathfrak{v}') \in |\mathcal{M}od_{\mathrm{PML}}(\Sigma')|$ *et toute* Σ*-formule* $\varphi \in \mathcal{S}en_{\mathrm{PML}}(\Sigma)$, $(\mathfrak{E}', \mathfrak{R}', \mathfrak{v}') \models^{\mathrm{PML}}_{\Sigma'} \mathcal{S}en_{\mathrm{PML}}(\sigma)(\varphi)$ *si et seulement si* $\mathcal{M}od_{\mathrm{PML}}(\sigma)((\mathfrak{E}', \mathfrak{R}', \mathfrak{v}')) \models^{\mathrm{PML}}_{\Sigma} \varphi$.

Preuve La preuve se fait par induction sur la structure des formules.
Notons tout d'abord que l'ensemble des états et la relation d'accessibilité sont tous deux préservés au travers du foncteur d'oubli $\mathcal{M}od(\sigma)$. Par définition, pour toute variable propositionnelle $p \in \Sigma$, on a $\mathfrak{v}(p) = \mathfrak{v}'(\sigma(p))$ ce qui prouve la condition de satisfaction pour les variables propositionnelles. La préservation de la condition de satisfaction par les connecteurs logiques et la modalité se vérifie simplement d'après les définitions. □

Exemple 1.1.3 (logique du 1^{er} ordre multi-sorte avec égalité)
L'institution FOL *de la logique du* 1^{er} *ordre multi-sorte avec égalité est définie de la façon suivante (cf. annexe B pour une présentation plus détaillée des notions de morphismes de signatures, de termes, de formules [atomiques et générales], de morphismes de structures du premier ordre et d'interprétations des variables) :*

- $\mathcal{S}ig_{\mathrm{FOL}}$ *est la catégorie des* signatures du premier ordre. *On rappelle qu'une signature du premier ordre est la donnée d'un triplet* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$, *où* $\mathcal{S}$ *est un ensemble de noms de sortes ;* $\mathcal{F} = (\mathcal{F}_{\omega,s})_{(\omega,s)\in\mathcal{S}^*\times\mathcal{S}}$ *est une famille d'ensembles de noms de fonctions (on notera* $f_{\omega\to s}$ *le fait que* $f \in \mathcal{F}_{\omega,s}$*) ;* $\mathcal{R} = (\mathcal{R}_\omega)_{\omega\in\mathcal{S}^*}$ *est une famille d'ensembles de noms de prédicats (ou relations) indexés par les arités. On rappelle également que les morphismes de signatures du premier ordre respectent la structure du triplet* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ *et préservent les arités et les sortes des symboles de fonctions et de prédicats ;*
- $\mathcal{S}en_{\mathrm{FOL}} : \mathcal{S}ig_{\mathrm{FOL}} \to \mathcal{S}et$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{FOL}}|$ *l'ensemble* $\mathcal{S}en_{\mathrm{FOL}}(\Sigma) \in |\mathcal{S}et|$ *des* Σ-formules du premier ordre *closes, i.e. l'ensemble construit au-dessus de l'ensemble des* Σ*-formules atomiques*[5] *à l'aide de la négation* ($\neg$)*, de la disjonction* ($\vee$) *et du quantificateur universel* ($\forall$)*, et répercute les changements de signatures sur les formules ;*
- $\mathcal{M}od_{\mathrm{FOL}} : \mathcal{S}ig_{\mathrm{FOL}} \to \mathcal{C}at$ *est un foncteur qui associe à toute signature* $(\mathcal{S}, \mathcal{F}, \mathcal{R}) \in |\mathcal{S}ig_{\mathrm{FOL}}|$ *la catégorie* $\mathcal{M}od_{\mathrm{FOL}}((\mathcal{S}, \mathcal{F}, \mathcal{R})) \in |\mathcal{C}at|$ *des* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-structures du premier ordre. *On rappelle qu'une* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$*-structure du premier ordre* $\mathcal{M}$ *est la donnée d'un ensemble de base* M_s *pour chaque symbole de sorte* $s \in \mathcal{S}$*, d'une fonction* $f^{\mathcal{M}} : M_{s_1} \times \cdots \times M_{s_n} \to$

5. On rappelle que les Σ-formules atomiques sont soit des égalités de la forme $t = t'$ où $t, t' \in T_\Sigma(\mathcal{X})$ sont des Σ-termes de même sorte, soit des formules prédicatives de la forme $r(t_1, \ldots, t_n)$ où $r \in \mathcal{R}_{s_1\ldots s_n}$ est un symbole de prédicat d'arité $s_1 \ldots s_n$ et $t_1, \ldots, t_n \in T_\Sigma(\mathcal{X})$ sont des termes, respectivement de sortes $s_1, \ldots, s_n$.

M_s pour chaque symbole de fonction $f \in \mathcal{F}_{s_1\dots s_n,s}$ et d'une relation $r^{\mathcal{M}} \subseteq M_{s_1} \times \cdots \times M_{s_n}$ pour chaque symbole de relation $r \in \mathcal{R}_{s_1\dots s_n}$. On rappelle qu'un morphisme de $(\mathcal{S},\mathcal{F},\mathcal{R})$-structures est un homomorphisme de $(\mathcal{S},\mathcal{F},\mathcal{R})$-structures. Étant donnés un morphisme de signatures $\sigma \in \mathcal{S}ig_{\mathrm{FOL}}(\Sigma,\Sigma')$, et une Σ'-structure $\mathcal{M}' \in |\mathcal{M}od_{\mathrm{FOL}}(\Sigma')|$, le réduit de $\mathcal{M}'$ est la Σ-structure $\mathcal{M}od_{\mathrm{FOL}}(\sigma)(\mathcal{M}')$ telle que :

- $\mathcal{M}od_{\mathrm{FOL}}(\sigma)(M')_s = M'_{\sigma_{\mathrm{SORTES}}(s)}$ *pour tout symbole de sorte* $s \in \mathcal{S}$,
- $f^{\mathcal{M}od_{\mathrm{FOL}}(\sigma)(\mathcal{M}')} = \sigma_{\mathrm{FONCTIONS}}(f)^{\mathcal{M}'}$ *pour tout symbole de fonction* $f \in \mathcal{F}$,
- $r^{\mathcal{M}od_{\mathrm{FOL}}(\sigma)(\mathcal{M}')} = \sigma_{\mathrm{RELATIONS}}(r)^{\mathcal{M}'}$ *pour tout symbole de relation* $r \in \mathcal{R}$.

- *la relation de satisfaction entre les modèles et les formules d'une même signature est stratifiée, pour chaque modèle, par les interprétations des variables. De ce fait,* $\models_{\mathrm{FOL}} = (\models^{\mathrm{FOL}}_{\Sigma})_{\Sigma \in |\mathcal{S}ig_{\mathrm{FOL}}|}$ *est une famille de relations binaires telle que pour toute signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{FOL}}|$, *la relation* $\models^{\mathrm{FOL}}_{\Sigma} \subseteq |\mathcal{M}od_{\mathrm{FOL}}| \times \mathcal{S}en_{\mathrm{FOL}}(\Sigma)$ *est définie de la façon suivante (où* $\mathcal{M} \in |\mathcal{M}od_{\mathrm{FOL}}(\Sigma)|$ *et* $\nu : (\mathcal{X}_s)_{s\in\mathcal{S}} \to (M_s)_{s\in\mathcal{S}}$ *est une interprétation des variables de* $\mathcal{X}$*) :*
  - $\forall t,t' \in T_\Sigma(\mathcal{X})$, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} t = t'$ *ssi* $\nu(t) = \nu(t')$,
  - $\forall r \in \mathcal{R}, \forall t_1,\dots,t_n \in T_\Sigma(\mathcal{X})$, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} r(t_1,\dots,t_n)$ *ssi*
  $$(\nu(t_1),\dots,\nu(t_n)) \in r^{\mathcal{M}}$$
  - $\forall \varphi \in \mathcal{S}en_{\mathrm{FOL}}(\Sigma)$, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} \neg\varphi$ *ssi* $\mathcal{M} \not\models^{\mathrm{FOL}}_{\Sigma,\nu} \varphi$,
  - $\forall x \in \mathcal{X}, \forall \varphi \in \mathcal{S}en_{\mathrm{FOL}}(\Sigma)$, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} (\forall x)\varphi$ *ssi* $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu'} \varphi$ *pour toute interprétation* ν' *tel que* $\nu'(y) = \nu(y)$ *pour toute variable* $y \in \mathcal{X}$, *sauf éventuellement pour* x,
  - $\forall \varphi,\psi \in \mathcal{S}en_{\mathrm{FOL}}(\Sigma)$, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} \varphi \vee \psi$ *ssi* $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} \varphi$ *ou* $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} \psi$

  Alors, $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma} \varphi$ *si pour toute interprétation de variables* $\nu : \mathcal{X} \to M$ *on a* $\mathcal{M} \models^{\mathrm{FOL}}_{\Sigma,\nu} \varphi$.

Exemple 1.1.4 (logique du 1^{er} ordre partielle multi-sorte avec égalité) *L'institution* PFOL *de la logique partielle multi-sorte avec égalité est obtenue à partir de la logique du premier ordre en introduisant des symboles de fonctions partielles. Une signature du premier ordre partielle est alors un quadruplet* $(\mathcal{S},\mathcal{TF},\mathcal{PF},\mathcal{R})$, *où* $\mathcal{TF}$ *est l'ensemble des symboles de fonctions totales et* $\mathcal{PF}$ *est l'ensemble des symboles de fonctions partielles. Une structure du premier ordre partielle est une structure du premier ordre ordinaire qui interprète les symboles de fonctions partielles par des fonctions partielles. Les formules atomiques sont de trois formes différentes :* défini *($\mathcal{D}ef$), égalité forte ($\stackrel{s}{=}$) et égalité existentielle ($\stackrel{e}{=}$). La formule $\mathcal{D}ef(t)$, où t est un terme, est vérifiée dans une structure du premier ordre partielle $\mathcal{M}$ si l'interprétation de t dans $\mathcal{M}$ est définie. L'égalité forte $t \stackrel{s}{=} t'$ est vérifiée lorsque les deux termes t et t' sont soit indéfinis, soit tous deux définis et égaux. L'égalité existentielle $t \stackrel{e}{=} t'$ est vérifiée lorsque les deux termes t et t' sont tous deux définis et égaux. Les formules sont alors les formules du premier ordre usuelles construites sur les formules atomiques définies ci-dessus.*

Exemple 1.1.5 (PFOL avec sous-sortes) *L'institution* SUBPFOL *de la logique partielle multi-sorte avec égalité et sous-sortes est obtenue à partir de* *la logique du* 1^{er} *ordre partielle multi-sorte avec égalité en introduisant la notion de sous-sorte. Une signature du premier ordre partielle avec sous-sortes est alors une signature du premier ordre partielle* $(\mathcal{S}, \mathcal{TF}, \mathcal{PF}, \mathcal{R})$ *munie d'une relation de pré-ordre sur les noms de sortes. On considère alors deux familles de fonctions supplémentaires traduisant les notions d'inclusion implicite de sortes* $em_{s_1,s_2} : s_1 \to s_2$ *et de projection partielle explicite* $pr_{s_1,s_2} : s_1 \to s_2$*. Les formules sont les formules de* PFOL *auxquelles on ajoute les formules construites sur le prédicat d'appartenance* $in_{s_1,s_2} \subseteq \mathcal{S} \times \mathcal{S}$*.*

Les institutions PFOL et SUBPFOL définies ci-dessus sont des institutions sous-jacentes aux langages de spécification CASL (*cf.* [99, 103]).

Exemple 1.1.6 (logique équationnelle) *L'institution* EQL *de la logique équationnelle est obtenue à partir de la logique du premier ordre en laissant de côté les symboles de relations et leur interprétation. La catégorie* $\mathcal{S}ig_{\mathrm{EQL}}$ *des signatures est donc la catégorie des* signatures algébriques *munie des morphismes de signatures algébriques et, pour toute signature* $\Sigma \in |\mathcal{S}ig_{\mathrm{EQL}}|$, $\mathcal{M}od_{\mathrm{EQL}}(\Sigma)$ *est la catégorie des* Σ-algèbres *munie des morphismes de* Σ*-algèbres (cf. annexe B).*

Exemple 1.1.7 (logique des clauses de Horn) *Une* formule de Horn universelle *pour une signature du premier ordre* $(\mathcal{S}, \mathcal{F}, \mathcal{R}) \in |\mathcal{S}ig_{\mathrm{FOL}}|$ *est une formule de la forme* $(\forall X)H \Rightarrow C$*, où* X *est un ensemble de variables,* H *est une conjonction finie de formules prédicatives ou équationnelles et* C *est une formule prédicative ou équationnelle. L'institution* HCL *de la logique des clauses de Horn est donc définie de la même façon que la logique du premier ordre mais l'ensemble de formules associé à une signature donnée est restreint aux formules de Horn universelles.*

Exemple 1.1.8 (logique relationnelle) *L'institution* REL *de la logique relationnelle est la sous-institution de* FOL *définie par les signatures ne contenant pas de symbole de fonction non-constante.*

Remarque 1.1.1 *L'institution* PL *de la logique propositionnelle peut également être obtenue à partir de l'institution* FOL *de la logique du premier ordre en restreignant les signatures à celles dont l'ensemble de noms de sortes est réduit à un singleton et l'ensemble de noms de fonctions est vide.*

Exemple 1.1.9 (logique du second ordre) *L'institution* SOL *de la logique du second ordre est l'extension de l'institution* FOL *admettant la quantification sur les noms de sortes, de fonctions et de relations.*

Exemple 1.1.10 (algèbre partielle) *L'institution* PA *des algèbres partielles raffine l'institution* EQL *en introduisant des symboles de fonctions partielles. Elle est obtenue à partir de* EQL *de la même façon que* PFOL *est obtenue à partir de* EQL*.*
Une signature algébrique partielle *est donc un triplet* $(\mathcal{S}, \mathcal{TF}, \mathcal{PF})$*, où* $\mathcal{TF}$ *est l'ensemble des symboles de fonctions totales et* $\mathcal{PF}$ *est l'ensemble des symboles de fonctions partielles. Une* algèbre partielle *est une algèbre ordinaire qui interprète les symboles de fonctions partielles par des fonctions partielles. Les formules atomiques sont de trois formes différentes :* défini *(*$\mathcal{D}ef$*), égalité forte (*$\stackrel{s}{=}$*) et égalité existentielle (*$\stackrel{e}{=}$*). Les formules sont alors les formules équationnelles usuelles construites sur les formules atomiques définies ci-dessus.*

Exemple 1.1.11 (automates) *Étant donné un ensemble V (de symboles d'entrées), un V-automate A consiste en la donnée :*

- *un ensemble $\mathfrak{A}$ d'états avec un état distingué $a_0 \in \mathfrak{A}$ appelé état initial et un sous-ensemble distingué d'états finaux ;*
- *une fonction de transition $T : V \times \mathfrak{A} \to \mathfrak{A}$.*

Un homomorphisme de V-automate $h : A \to B$ consiste en la donnée d'une fonction $h : \mathfrak{A} \to \mathfrak{B}$ telle que $h(a_0) = b_0$, $h(a)$ est final seulement si a est final et le diagramme ci-dessous commute.

$$\begin{array}{ccc} V \times \mathfrak{A} & \xrightarrow{T_A} & \mathfrak{A} \\ {\scriptstyle \mathtt{Id}_V \times \mathtt{h}} \downarrow & & \downarrow {\scriptstyle h} \\ V \times \mathfrak{B} & \xrightarrow[T_B]{} & \mathfrak{B} \end{array}$$

La fonction de transition s'étend canoniquement par itération aux mots sur V et un mot $w \in V^*$ est reconnu par un V-automate si et seulement si $T(w, a_0)$ est un état final.
L'institution AUT *des automates a pour catégorie de signatures la catégorie $\mathcal{S}et$ des ensembles, les automates comme modèles et les chaînes de symboles d'entrée pour formules. Une chaîne w est satisfaite par un automate A si A reconnaît w.*

1.2 Logique interne

La définition des institutions n'implique aucune contrainte sur la construction et la forme des formules. La seule chose imposée est que les formules associées à une signature Σ donnée forment un ensemble auquel il est possible d'étendre tout morphisme de signatures impliquant Σ sans altérer la satisfaction des formules par les modèles associés à Σ et leurs réduits.
Dans la pratique cependant, la structure des formules n'est pas quelconque. Comme on le voit sur les exemples donnés dans la section précédente, la définition des formules pour une logique particulière repose sur la donnée d'un ensemble d'éléments de base (les formules atomiques) et d'opérateurs sur cet ensemble (les connecteurs logiques, quantificateurs, modalités, *etc.*). Pour une même signature, les ensembles de formules peuvent alors être très différents suivant l'ensemble de constructeurs considéré (voir par exemple les différences entre les ensembles de formules de la logique des clauses de Horn et de la logique des prédicats du premier ordre). Or, de nombreux résultats plus ou moins généraux en théorie de la spécification et en théorie des modèles font l'hypothèse de l'existence de certains de ces opérateurs. Par exemple, l'équivalence entre la propriété d'interpolation de Craig et le lemme de consistance de Robinson suppose la négation (*cf.* chapitre 2). Il est donc indispensable de pouvoir définir des notions telles que celle de connecteur logique ou de quantification de façon interne à une institution quelconque.

La définition institution-indépendante de la plupart des connecteurs logiques usuels est assez simple comme le montre la définition suivante.

Notation 1.2.1 *Soit $E \in |\mathcal{Set}|$ un ensemble. On note $\wp(E)$ l'ensemble des parties de E, i.e. :*

$$\wp(E) = \{S / S \subseteq E\}$$

Si E est un ensemble infini, on note $\wp_f(E)$ l'ensemble des parties finies de E.

Définition 1.2.1 (connecteurs logiques (sémantiques)) *Soit une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ et soit une signature $\Sigma \in |\mathcal{Sig}|$.*

négation : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est la* négation (sémantique) *d'une formule $\varphi \in \mathcal{Sen}(\Sigma)$ si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \mathcal{M} \not\models_\Sigma \varphi$$

On notera alors $\neg\varphi$ cette formule.

conjonction (finie) : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est la* conjonction (sémantique) finie *d'un ensemble fini de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp_f(\mathbb{N})$) si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \forall i \in I,\ \mathcal{M} \models_\Sigma \varphi_i$$

On notera alors $\bigwedge_{i \in I} \varphi_i$ cette formule.

conjonction (infinie) : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est la* conjonction (sémantique) infinie *d'un ensemble infini de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp(\mathbb{N})$) si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \forall i \in I,\ \mathcal{M} \models_\Sigma \varphi_i$$

On notera alors $\bigwedge_{i \in I} \varphi_i$ cette formule.

disjonction (finie) : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est la* disjonction (sémantique) finie *d'un ensemble fini de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp_f(\mathbb{N})$) si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \exists i \in I,\ \mathcal{M} \models_\Sigma \varphi_i$$

On notera alors $\bigvee_{i \in I} \varphi_i$ cette formule.

disjonction (infinie) : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est la* disjonction (sémantique) infinie *d'un ensemble infini de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp(\mathbb{N})$) si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \exists i \in I,\ \mathcal{M} \models_\Sigma \varphi_i$$

On notera alors $\bigvee_{i \in I} \varphi_i$ cette formule.

implication : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est l'*implication (sémantique) *d'une Σ-formule $\varphi_2 \in \mathcal{Sen}(\Sigma)$ par une Σ-formule $\varphi_1 \in \mathcal{Sen}(\Sigma)$ si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow (\mathcal{M} \models_\Sigma \varphi_1 \Rightarrow \mathcal{M} \models_\Sigma \varphi_1)$$

On notera alors $\varphi_1 \Rightarrow \varphi_2$ cette formule.

équivalence : *une Σ-formule $\varphi' \in \mathcal{Sen}(\Sigma)$ est l'*équivalence (sémantique) *des Σ-formules $\varphi_1, \varphi_2 \in \mathcal{Sen}(\Sigma)$ si et seulement si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M} \models_\Sigma \varphi' \Leftrightarrow (\mathcal{M} \models_\Sigma \varphi_1 \Leftrightarrow \mathcal{M} \models_\Sigma \varphi_1)$$

On notera alors $\varphi_1 \Leftrightarrow \varphi_2$ cette formule.

Remarque 1.2.1 *On notera que cette définition des connecteurs logiques n'est en fait valable que pour les formules closes. Le fait qu'elle ne le soit pas pour les formules ouvertes est du à l'interprétation universelle des variables libres contenues dans les formules ouvertes. Ainsi pour la négation par exemple, il peut exister des modèles ne satisfaisant ni la formule φ ni sa négation, i.e. φ est vraie uniquement pour certaines interprétations des variables et sa classe de modèles n'est donc pas le complémentaire de celle de $\neg\varphi$.*

r

Définition 1.2.2 (équivalence sémantique) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Soit $\Sigma \in |\mathcal{Sig}|$ une signature et soient $\varphi_1, \varphi_2 \in \mathcal{Sen}(\Sigma)$ des Σ-formules. On dit que φ_1 et φ_2 sont* sémantiquement équivalentes *si, pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, on a :*

$$\mathcal{M} \models_\Sigma \varphi_1 \Leftrightarrow \mathcal{M} \models_\Sigma \varphi_2$$

Proposition 1.2.1 *La négation, la conjonction (finie et infinie), la disjonction (finie et infinie) de formules sont uniques modulo l'équivalence sémantique. Il en est de même pour l'implication et l'équivalence.*

Preuve La preuve découle directement de la définition des connecteurs logiques et de l'équivalence sémantique. □

Définition 1.2.3 (fermeture) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. On dit que $\mathcal{I}$ est fermée par (ou possède la) :*

- négation (sémantique) *si pour toute signature $\Sigma \in |\mathcal{Sig}|$, toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ possède une négation (sémantique). On notera $\neg\varphi$ une telle formule ;*
- conjonction (sémantique) finie *(resp. infinie) si pour toute signature $\Sigma \in |\mathcal{Sig}|$, tout ensemble fini (resp. infini) de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp_f(\mathbb{N})$ [resp. $I \in \wp(\mathbb{N})$]), possède une conjonction (sémantique) finie (resp. infinie). On notera $\bigwedge_{i \in I} \varphi_i$ une telle formule ;*
- disjonction (sémantique) finie *(resp. infinie) si pour toute signature $\Sigma \in |\mathcal{Sig}|$, tout ensemble fini (resp. infini) de Σ-formules $\{\varphi_i / i \in I\} \subseteq \mathcal{Sen}(\Sigma)$ ($I \in \wp_f(\mathbb{N})$ [resp. $I \in \wp(\mathbb{N})$]), possède une disjonction (sémantique) finie (resp. infinie). On notera $\bigvee_{i \in I} \varphi_i$ une telle formule ;*
- implication (resp. équivalence) *si et seulement si pour toute signature $\Sigma \in |\mathcal{Sig}|$, tout couple de Σ-formules $(\varphi_1, \varphi_2) \in \mathcal{Sen}(\Sigma) \times \mathcal{Sen}(\Sigma)$ est une implication (resp. équivalence). On notera $\varphi_1 \Rightarrow \varphi_2$ (resp. $\varphi_1 \Leftrightarrow \varphi_2$) une telle formule.*

De par la définition 1.2.1, il est clair que l'institution des prédicats du premier ordre est fermée par négation.

Il est clair que la traduction sémantique des connecteurs logiques donnée ci-dessus est transportée le long des morphismes de signatures.

Proposition 1.2.2 *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Soit un morphisme de signatures $\sigma \in \mathcal{Sig}(\Sigma_1, \Sigma_2)$ et soit $(\varphi_1, \ldots, \varphi_n) \in \mathcal{Sen}(\Sigma_1)^n$ un n-uplet de Σ_1-formules ($n \in \mathbb{N} \setminus \{0\}$). Alors, pour tout connecteur logique $\top \in \{\neg, \vee, \wedge, \Rightarrow, \Leftrightarrow\}$, on a $\mathcal{Sen}(\sigma)(\top(\varphi_1, \ldots, \varphi_n)) = \top(\mathcal{Sen}(\sigma)(\varphi_1), \ldots, \mathcal{Sen}(\sigma)(\varphi_n))$.*

Preuve Ceci découle directement de la définition des connecteurs logiques et de la condition de satisfaction. □

La définition institution-indépendante de la quantification est légèrement plus complexe. Elle s'appuie sur les notions de morphisme de signatures et d'expansion de modèles le long de ces morphismes. Intuitivement, une interprétation de variables dans un modèle $\mathcal{M}$ est simplement une expansion de $\mathcal{M}$ le long d'un morphisme particulier de signatures. Prenons le cas de la logique du premier ordre par exemple. Soit $\mathcal{X}$ un ensemble de variables pour la signature $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ et soit $\forall\mathcal{X}\varphi$ une formule universellement quantifiée. En considérant $\mathcal{X}$ comme un ensemble de constantes, remarquons que pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{M} \models_\Sigma \forall\mathcal{X}\varphi$ si et seulement si $\mathcal{M}' \models_{\Sigma'} \varphi$ pour toute expansion [6] de $\mathcal{M}$ le long d'un morphisme de signatures $\Sigma \to \Sigma'$, où $\Sigma' = (\mathcal{S}, \mathcal{F} \cup \mathcal{X}, \mathcal{R})$. Ainsi, la satisfaction des formules quantifiées peut être définie en termes d'inclusion de signatures (ou plus généralement de morphismes de signatures) et de réduction de modèles, qui sont des concepts institution-indépendants, plutôt qu'en termes de variables et d'interprétations de variables qui sont des concepts institution-dépendants.

Définition 1.2.4 (quantification universelle (sémantique) [43]) *Soit une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ et soit $\chi : \Sigma \to \Sigma'$ un morphisme de signatures. Une Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ est la χ-*quantification universelle (sémantique) *de la Σ'-formule $\varphi' \in \mathcal{Sen}(\Sigma')$ si et seulement si pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, on a :*

$$\mathcal{M} \models_\Sigma \varphi \Leftrightarrow (\forall \mathcal{M}' \in |\mathcal{Mod}(\Sigma')|, \mathcal{Mod}(\sigma)(\mathcal{M}') = \mathcal{M} \Rightarrow \mathcal{M}' \models_{\Sigma'} \varphi')$$

On notera $(\forall\chi)\varphi'$ la Σ-formule φ.

Définition 1.2.5 (quantification existentielle (sémantique) [43]) *Soit une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ et soit $\chi : \Sigma \to \Sigma'$ un morphisme de signatures. Une Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ est la χ-*quantification existentielle (sémantique) *de la Σ'-formule $\varphi' \in \mathcal{Sen}(\Sigma')$ si et seulement si pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, on a :*

$$\mathcal{M} \models_\Sigma \varphi \Leftrightarrow \exists \mathcal{M}' \in |\mathcal{Mod}(\Sigma')|, \mathcal{Mod}(\sigma)(\mathcal{M}') = \mathcal{M} \wedge \mathcal{M}' \models_{\Sigma'} \varphi'$$

On notera $(\exists\chi)\varphi'$ la Σ-formule φ.

Notation 1.2.2 *Soient $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution et $\mathcal{O}$ un ensemble d'opérateurs sur les formules (i.e. des connecteurs logiques ou des quantificateurs). Soit $E \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de formules construites sur une signature $\Sigma \in |\mathcal{Sig}|$. On note $\mathcal{O}(E)$ le plus petit ensemble de formules (au sens de l'inclusion) contenant E et fermée par (les éléments de) $\mathcal{O}$.*

On notera qu'on n'impose nulle part que l'institution $\mathcal{I}$ possède toutes les formules de $\mathcal{O}(E)$.

Définition 1.2.6 (accessibilité) *Soient $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution et $\mathcal{O}$ un ensemble d'opérateurs sur les formules. Soient $\Sigma \in |\mathcal{Sig}|$ une signature et $E \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de Σ-formules. Une Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ est dite* accessible à partir de E par O *si φ est sémantiquement équivalente à une formule de $\mathcal{O}(E)$.*

6. Dans le cas de la quantification existentielle, on remplacera « pour toute expansion » par « pour une expansion ».

Exemple 1.2.1 *Toute formule de l'institution de la logique des prédicats du premier ordre (cf. exemple 1.1.3) est accessible à partir des atomes par* $\neg$, $\wedge$ *et la quantification universelle par extension de signature à un nombre fini de constantes.*

Exemple 1.2.2 *Toute formule de l'institution de la logique équationnelle (cf. exemple 1.1.6) est accessible à partir des équations par la quantification universelle de l'exemple précédent.*

Exemple 1.2.3 *Toute formule de l'institution de la logique des clauses de Horn (cf. exemple 1.1.7) est accessible à partir des atomes par* $\wedge$*, l'implication et la quantification universelle de l'exemple 1.2.1. On notera que, pour toute signature* $\Sigma \in |\mathit{Sig}_{\mathrm{HCL}}|$*, l'ensemble* $\mathit{Sen}_{\mathrm{HCL}}(\Sigma)$ *des* Σ*-formules de l'institution de la logique des clauses de Horn est strictement inclus dans* $\mathcal{O}(\mathit{Atom}_\Sigma)$*, où* $\mathcal{O} = \{\wedge, \Rightarrow, \forall\}$ *et* Atom_Σ *est l'ensemble des* Σ*-formules atomiques, i.e. les équations entre* Σ*-termes et les* Σ*-formules prédicatives.*

Parce qu'elle possède des propriétés modèle-théoriques particulières, la quantification du premier ordre joue un rôle important en théorie standard des modèles. Ainsi que le montrent T. Mossakowski, J. Goguen, R. Diaconescu et A. Tarlecki dans [39], il est possible de donner une caractérisation institution-indépendante de la notion de variable du premier ordre. Intuitivement, étant donné un ensemble $\mathcal{X}$ de variables du premier ordre, les interprétations de $\mathcal{X}$ dans un $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-modèle $\mathcal{M}$ sont en bijection canonique avec les homomorphismes de $\mathcal{F}$-algèbres $T_{\mathcal{F}}(\mathcal{X}) \to \mathcal{M}$, où $T_{\mathcal{F}}(\mathcal{X})$ est la $\mathcal{F}$-algèbre de termes générée librement au-dessus de $\mathcal{X}$. De ce fait, la donnée d'un $(\mathcal{S}, \mathcal{F} \cup \mathcal{X}, \mathcal{R})$-modèle est la même chose que la donnée d'un morphisme de $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-modèles de domaine $T_{\mathcal{F}}(\mathcal{X})$. Cette propriété catégorielle des inclusions de signatures est la base pour le traitement institution-indépendant des variables du premier ordre. On définit alors le concept de *morphisme de signatures représentable.*

Définition 1.2.7 (morphisme représentable) *Soit* $\mathcal{I} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, \models)$ *une institution. Un morphisme* $\sigma \in \mathit{Sig}(\Sigma, \Sigma')$ *est dit* représentable *s'il existe un* Σ*-modèle* $\mathcal{M}_\sigma$ *(appelé la* représentation de σ*) et un isomorphisme* i_σ *de catégories tels que le diagramme suivant soit commutatif (où* $\mathcal{M}_\sigma/\mathit{Mod}(\Sigma)$ *est la catégorie virgule dont les objets sont les morphismes de* Σ*-modèles de domaine* $\mathcal{M}$ *(cf. annexe A)).*

$$\begin{tikzcd}
\mathit{Mod}(\Sigma') \arrow[r, "i_\sigma"] \arrow[dr, "\mathit{Mod}(\sigma)"'] & \mathcal{M}_\sigma/\mathit{Mod}(\Sigma) \arrow[d, "\text{oubli}"] \\
 & \mathit{Mod}(\Sigma)
\end{tikzcd}$$

Notons que les morphismes représentables de signatures capturent exactement le concept de variables du premier ordre. Cette caractérisation est de plus institution-indépendante. Les extensions de signatures avec des variables d'ordre supérieur (tel que les symboles de sortes, de fonctions et de prédicats) ne sont pas des morphismes représentables de signatures. En fait, on montre même la caractérisation suivante des morphismes représentables de signatures dans l'institution des prédicats du premier ordre.

Proposition 1.2.3 ([43]) *Un morphisme de signatures du premier ordre est représentable si et seulement s'il est bijectif sur les symboles de sortes, de relations et de fonctions non-constantes.*

1.3 Présentations et théories

Comme nous l'avons dit en introduction, l'une des motivations pour l'introduction des institutions est la généralisation de résultats en théorie de la spécification. Intuitivement, une spécification axiomatique est essentiellement la donnée d'une *signature* et d'un ensemble de formules construites sur cette signature, souvent appelées *axiomes*, et décrivant les propriétés que l'on attend du logiciel en spécification. Ce type de construction se généralise aisément dans le cadre de la théorie des institutions. Intuitivement, une spécification est alors la donnée d'un ensemble de formules construites sur une signature donnée. On parle alors de Σ-*présentation* ou de Σ-*spécification à plat* (où Σ désigne la signature sur laquelle sont construites les formules de la présentation). Ce qui est intéressant ici est qu'il est possible de capturer le concept de spécification et de démontrer un certain nombre de résultats standards sur les spécifications indépendamment des détails du système logique sous-jacent présenté par une institution, *i.e.* sans connaissance de la composition des signatures, de la structure des modèles ou des formules ou encore de la définition exacte de la relation de satisfaction. En fait, seule la condition de satisfaction est utile pour obtenir les résultats souhaités.
Dans toute la suite, nous supposerons donnée une institution arbitraire fixe $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$.

Notation 1.3.1 *Soit $\Sigma \in |\mathcal{Sig}|$. Pour tout ensemble de Σ-formules $\Phi \subseteq \mathcal{Sen}(\Sigma)$ et tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, on notera $\mathcal{M} \models_\Sigma \Phi$ si $\mathcal{M} \models_\Sigma \varphi$ pour toute Σ-formule $\varphi \in \Phi$.*

Définition 1.3.1 *Soit $\Sigma \in |\mathcal{Sig}|$ et soient $\Phi \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de Σ-formules et $\mathscr{M} \subseteq |\mathcal{Mod}(\Sigma)|$ une collection de Σ-modèles.*

1. *On note $\mathcal{Mod}(\Phi)$ la* collection de tous les Σ-modèles satisfaisant Φ, *i.e. :*

$$\mathcal{Mod}(\Phi) = \{\mathcal{M} \in |\mathcal{Mod}(\Sigma)| / \mathcal{M} \models_\Sigma \Phi\}$$

2. *On note $\mathcal{Th}(\mathscr{M})$ la* collection de toutes les Σ-formules satisfaites par tous les Σ-modèles appartenant à $\mathscr{M}$, *i.e. :*

$$\mathcal{Th}(\mathscr{M}) = \{\varphi \in \mathcal{Sen}(\Sigma) / \forall \mathcal{M} \in \mathscr{M},\ \mathcal{M} \models_\Sigma \varphi\}$$

$\mathcal{Th}(\mathscr{M})$ *est appelée la* théorie de $\mathscr{M}$.

3. *De façon similaire, on note $\mathcal{Th}(\Phi)$ la* collection de toutes les Σ-formules satisfaites par tous les Σ-modèles de Φ, *i.e. :*

$$\mathcal{Th}(\Phi) = \{\varphi \in \mathcal{Sen}(\Sigma) / \forall \mathcal{M} \in |\mathcal{Mod}(\Phi)|,\ \mathcal{M} \models_\Sigma \varphi\} = \mathcal{Th}(\mathcal{Mod}(\Phi))$$

$\mathcal{Th}(\Phi)$ *est appelée* fermeture de Φ.

4. *On note* $\models_\Sigma$ *la relation définie sur* $\wp(\mathcal{S}en(\Sigma)) \times \mathcal{S}en(\Sigma)$ *de la façon suivante :*

$$\forall \varphi \in \mathcal{S}en(\Sigma),\ \Phi \models_\Sigma \varphi \Leftrightarrow \varphi \in \mathcal{T}h(\Phi)$$

   On dit alors que φ *est une* conséquence sémantique *de* Φ.
5. *Si* $\Phi = \mathcal{T}h(\Phi)$ *alors* Φ *est dit* fermé *ou* clos.
6. *Une* Σ-théorie T *est un ensemble fermé de* Σ*-formules, i.e.* $T = \mathcal{T}h(T)$. *On dit que* Φ présente *la théorie* T *si* T *est la théorie obtenue par fermeture de* Φ*, i.e. si* $T = \mathcal{T}h(\Phi)$.

Définition 1.3.2 (présentation) *Soit* $\Sigma \in |\mathcal{S}ig|$ *une signature. On appelle* Σ-présentation *(ou* Σ-spécification à plat*) la donnée d'un ensemble de* Σ*-formules* $P \subseteq \mathcal{S}en(\Sigma)$. *On appelle* Σ-axiome *toute formule* $\varphi \in P$.

Une présentation P *présente* la théorie $\mathcal{T}h(P)$. Dans la suite, nous parlerons indifféremment d'ensembles de formules ou de présentations.

Définition 1.3.3 (morphisme de présentations) *Soient* P *une* Σ*-présentation et* P' *une* Σ'*-présentation. Un* morphisme de présentations $\sigma : P \to P'$ *est un morphisme de signatures* $\sigma : \Sigma \to \Sigma'$ *tel que pour tout* Σ*-axiome* $\varphi \in P$*, son image par* $\mathcal{S}en(\sigma)$ *est une conséquence sémantique de* P'*, i.e.* $\mathcal{S}en(\sigma)(\varphi) \in \mathcal{T}h(P')$.

Un *morphisme de théories* est alors un morphisme de présentations $\sigma : P \to P'$ tel que P et P' sont des théories. Dans ce cas, on a $\mathcal{S}en(\sigma)(\Phi) \subseteq \Phi'$.

Proposition 1.3.1 *Soient* $\Phi, \Psi \subseteq \mathcal{S}en(\Sigma)$ *des ensembles de* Σ*-formules et soient* $\mathcal{M}, \mathcal{N} \subseteq |\mathcal{M}od(\Sigma)|$ *des classes de* Σ*-modèles.*
1. *Si* $\Phi \subseteq \Psi$ *alors* $\mathcal{M}od(\Psi) \subseteq \mathcal{M}od(\Phi)$.
2. *Si* $\mathcal{M} \subseteq \mathcal{N}$ *alors* $\mathcal{T}h(\mathcal{N}) \subseteq \mathcal{T}h(\mathcal{M})$.
3. $\Phi \subseteq \mathcal{T}h(\mathcal{M}od(\Phi))$ *et* $\mathcal{M} \subseteq \mathcal{M}od(\mathcal{T}h(\mathcal{M}))$.
4. $\mathcal{M}od(\Phi) = \mathcal{M}od(\mathcal{T}h(\mathcal{M}od(\Phi)))$ *et* $\mathcal{T}h(\mathcal{M}) = \mathcal{T}h(\mathcal{M}od(\mathcal{T}h(\mathcal{M})))$.

Preuve Immédiat d'après les définitions. □

Proposition 1.3.2 *La relation de conséquence sémantique est préservée par les morphismes de signatures, i.e. pour tout morphisme de signatures* $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$*, pour tout ensemble de* Σ*-formules* $\Phi \subseteq \mathcal{S}en(\Sigma)$ *et toute* Σ*-formule* $\psi \in \mathcal{S}en(\Sigma)$*, si* $\Phi \models_\Sigma \psi$ *alors* $\mathcal{S}en(\sigma)(\Phi) \models_{\Sigma'} \mathcal{S}en(\sigma)(\psi)$.

Preuve C'est une conséquence directe des définitions et de la condition de satisfaction. □

Proposition 1.3.3 *Soient* $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$ *un morphisme de signatures et* $\Phi' \subseteq \mathcal{S}en(\Sigma')$ *un ensemble clos de* Σ'*-formules. Soit* $\Phi'_\Sigma \subseteq \mathcal{S}en(\Sigma)$ *l'ensemble de* Σ*-formules dont l'image par* $\mathcal{S}en(\sigma)$ *appartient à* Φ'*, i.e. :*

$$\Phi'_\Sigma = \{\varphi \in \mathcal{S}en(\Sigma) / \mathcal{S}en(\sigma)(\varphi) \in \Phi'\}$$

L'ensemble Φ'_Σ *ainsi défini est un ensemble clos de* Σ*-formules.*

Preuve Il est clair d'après les définitions que $\Phi'_\Sigma \subseteq \mathcal{Th}(\Phi'_\Sigma)$. Il suffit donc de montrer que $\mathcal{Th}(\Phi'_\Sigma) \subseteq \Phi'_\Sigma$.
Soit donc $\varphi \in \mathcal{Th}(\Phi'_\Sigma)$. Par la proposition 1.3.2, $\mathcal{Sen}(\sigma)(\varphi)$ est alors une conséquence sémantique de $\mathcal{Sen}(\sigma)(\Phi'_\Sigma) \subseteq \Phi'$. Puisque Φ' est une théorie, ceci signifie que $\mathcal{Sen}(\sigma)(\varphi)$ appartient à Φ' et donc que φ appartient à Φ'_Σ. □

Proposition 1.3.4 *Soient $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$ un morphisme de signatures, $\Phi \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de Σ-formules et $\Phi' \subseteq \mathcal{Sen}(\Sigma')$ un ensemble de Σ'-formules. Les conditions suivantes sont équivalentes :*

1. *σ est un morphisme de théories de $(\Sigma, \mathcal{Th}(\Phi))$ dans $(\Sigma', \mathcal{Th}(\Phi'))$;*
2. *$\mathcal{Sen}(\sigma)(\Phi) \subseteq \mathcal{Th}(\Phi')$;*
3. *$\forall \mathcal{M}' \in \mathcal{Mod}(\Phi')$, $\mathcal{Mod}(\sigma)(\mathcal{M}') \in \mathcal{Mod}(\Phi)$.*

Preuve
$(1 \Rightarrow 2)$ Immédiat d'après les définitions.
$(2 \Rightarrow 3)$ Soit $\mathcal{M}' \in \mathcal{Mod}(\Phi')$. Par hypothèse, $\mathcal{Sen}(\sigma)(\Phi) \subseteq \mathcal{Th}(\Phi')$. On a donc $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma)(\Phi)$. Par la condition de satisfaction, on a alors $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_\Sigma \Phi$, *i.e.* $\mathcal{Mod}(\sigma)(\mathcal{M}') \in \mathcal{Mod}(\Phi)$.
$(3 \Rightarrow 1)$ Il suffit de montrer que $\mathcal{Sen}(\sigma)(\Phi) \subseteq \Phi'$. Soit $\varphi \in \Phi$. Par définition, pour tout $\mathcal{M} \in \mathcal{Mod}(\Phi)$, on a $\mathcal{M} \models_\Sigma \Phi$. En particulier, pour tout $\mathcal{M}' \in \mathcal{Mod}(\Phi')$, on a par hypothèse $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_\Sigma \Phi$. Par la condition de satisfaction, on a alors $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma)(\Phi)$. On en déduit alors $\mathcal{Sen}(\sigma)(\Phi) \subseteq \mathcal{Th}(\Phi')$, *i.e.* $\sigma : (\Sigma, \mathcal{Th}(\Phi)) \to (\Sigma', \mathcal{Th}(\Phi'))$ est un morphisme de théories.
□

Il est clair que les théories construites sur les signatures de l'institution $\mathcal{I}$ constituent les objets d'une catégorie, notée $\mathcal{Th}(\mathcal{I})$, dont les flèches sont les morphismes de théories. La proposition précédente nous permet de définir le foncteur $\mathcal{Mod}_{\mathcal{Th}(\mathcal{I})} : \mathcal{Th}(\mathcal{I}) \to \mathcal{Cat}^{op}$ qui, à toute théorie T, associe la sous-catégorie complète $\mathcal{Mod}(T) \subseteq \mathcal{Mod}(\Sigma)$ et, à tout morphisme de théories $\sigma : T \to T'$, associe le foncteur d'oubli $\mathcal{Mod}(\sigma) : \mathcal{Mod}(T') \to \mathcal{Mod}(T)$.

Définition 1.3.4 (consistance) *Soit $\Sigma \in |\mathcal{Sig}|$ une signature et soit $\Phi \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de Σ-formules. L'ensemble Φ est dit :*

- consistant *si $\mathcal{Th}(\Phi) \neq \mathcal{Sen}(\Sigma)$ et* inconsistant *dans le cas contraire ;*
- finiment consistant *si tout sous-ensemble fini $\Phi' \in \wp_f(\Phi)$ est consistant.*

Définition 1.3.5 (complétude) *Soit $\Sigma \in |\mathcal{Sig}|$ une signature et soit $T \subseteq \mathcal{Sen}(\Sigma)$ une Σ-théorie. La théorie T est dite* complète *(on dit aussi maximale consistante) si elle est consistante et si, pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$, l'une des deux conditions suivantes est vérifiée :*

- *$T \cup \{\varphi\}$ est inconsistante ;*
- *$\varphi \in T$.*

On suppose pour la suite de cette section une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ fermée par négation.

Proposition 1.3.5 *Soit $\Sigma \in |\mathcal{Sig}|$ une signature telle que $\mathcal{Sen}(\Sigma) = \varnothing$. Alors il n'existe pas de théorie complète sur Σ.*

Proposition 1.3.6 *Soit $\Sigma \in |\mathcal{Sig}|$ une signature telle que $\mathcal{Sen}(\Sigma) \neq \varnothing$. Une Σ-présentation P est inconsistante si et seulement si $\mathcal{Mod}(P) = \varnothing$.*

Preuve

($\Rightarrow$) Supposons que P soit inconsistante. Alors, la Σ-théorie présentée par P est l'ensemble de toutes les Σ-formules. Ceci signifie que pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$, φ et sa négation sont des conséquences sémantiques de P. Comme il n'existe pas de modèle satisfaisant à la fois φ et sa négation, on en déduit que $\mathcal{Mod}(P)$ est vide.

($\Leftarrow$) (par l'absurde) Supposons donc que $\mathcal{Mod}(P) = \varnothing$ et que P est consistante. Soit $\varphi \in \mathcal{Sen}(\Sigma)$. Nous allons montrer que φ est une conséquence sémantique de P. Puisque $\mathcal{Mod}(P) = \varnothing$, il est vrai que pour tout Σ-modèle $\mathcal{M} \in \mathcal{Mod}(P)$ on a $\mathcal{M} \models_\Sigma \varphi$. Aucune hypothèse n'ayant été faite sur φ on en déduit que ceci est vrai pour toutes les Σ-formules ce qui contredit le fait que P soit consistante.

□

Proposition 1.3.7 *Soit $\Sigma \in |\mathcal{Sig}|$ une signature telle que $\mathcal{Sen}(\Sigma) \neq \varnothing$. Pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{Th}(\mathcal{M})$ est une théorie complète.*

Preuve Il est clair que $\mathcal{Th}(\mathcal{M})$ est une Σ-théorie consistante d'après sa définition. Soit donc $\varphi \in \mathcal{Sen}(\Sigma) \setminus \mathcal{Th}(\mathcal{M})$. Par définition on a alors $\mathcal{M} \not\models_\Sigma \varphi$. Puisque $\mathcal{I}$ est fermée par négation, on a $\mathcal{M} \models_\Sigma \neg\varphi$, *i.e.* $\neg\varphi \in \mathcal{Th}(\mathcal{M})$. On en déduit que $\{\varphi\} \cup \mathcal{Th}(\mathcal{M})$ est une théorie inconsistante ce qui prouve que $\mathcal{Th}(\mathcal{M})$ est une Σ-théorie complète. □

Définition 1.3.6 (compacité) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. On dit que $\mathcal{I}$ est :*

- compacte *si pour tout ensemble de Σ-formules $\Phi \subseteq \mathcal{Sen}(\Sigma)$ ($\Sigma \in |\mathcal{Sig}|$), Φ est consistant si et seulement si Φ est finiment consistant ;*
- conséquence-compacte *si pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ et tout ensemble de Σ-formules $\Phi \subseteq \mathcal{Sen}(\Sigma)$, si $\Phi \models_\Sigma \varphi$ alors il existe un sous-ensemble fini $\Phi' \subseteq \Phi$ ($\Phi' \in \wp_f(\Phi)$) tel que $\Phi' \models_\Sigma \varphi$.*

Ces deux notions de compacité sont équivalentes pour toute institution fermée par négation comme le montre la proposition suivante.

Proposition 1.3.8 *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution fermée par négation. Alors $\mathcal{I}$ est compacte si et seulement si elle est conséquence-compacte.*

Preuve

($\Rightarrow$) Soient $\Phi \subseteq \mathcal{Sen}(\Sigma)$ un ensemble de Σ-formules et $\varphi \in \mathcal{Sen}(\Sigma)$ une Σ-formule tels que $\Phi \models_\Sigma \varphi$. On va montrer par l'absurde qu'il existe un sous-ensemble fini $\Phi' \in \wp_f(\Phi)$ tel que $\Phi' \models_\Sigma \varphi$.
Supposons donc le contraire, *i.e.* pour tout sous-ensemble fini $\Phi' \in \wp_f(\Phi)$, on a $\Phi' \not\models_\Sigma \varphi$. On en déduit alors que $\Phi' \cup \{\neg\varphi\}$ est consistant et donc que $\Phi \cup \{\neg\varphi\}$ est finiment consistant. Par hypothèse, $\Phi \cup \{\neg\varphi\}$ est donc consistant, *i.e.* $\Phi \not\models_\Sigma \varphi$ ce qui contredit les hypothèses.

($\Leftarrow$) Soit $\Phi \subseteq \mathcal{Sen}(\Sigma)$ un ensemble consistant de Σ-formules. Par la proposition 1.3.1 on montre $\mathcal{Th}(\Phi') \subseteq \mathcal{Th}(\Phi)$ pour tout $\Phi' \in \wp_f(\Phi)$. On en déduit que Φ' est consistant ce qui prouve que Φ est finiment consistant.
(par contraposée) Supposons maintenant que Φ soit inconsistant. Alors, pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$, on a $\Phi \models_\Sigma \varphi$ et $\Phi \models_\Sigma \neg\varphi$. Puisque Φ est conséquence-compacte, il existe deux sous-ensembles $\Phi', \Phi'' \in \wp_f(\Phi)$ tels que $\Phi' \models_\Sigma \varphi$ et $\Phi'' \models_\Sigma \neg\varphi$. Or, $\Phi' \cup \Phi'' \in \wp_f(\Phi)$ est un sous-ensemble fini de Φ et on a $\Phi' \cup \Phi'' \models_\Sigma \varphi$ et $\Phi' \cup \Phi'' \models_\Sigma \neg\varphi$. On en déduit alors que $\Phi' \cup \Phi''$ est inconsistant et donc que Φ est non finiment consistant.

□

1.4 Propriétés sémantiques

Définition 1.4.1 (amalgation) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Le diagramme commutatif de morphismes de signatures ci-dessous*

$$\begin{tikzcd}
\Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma'_1"] \\
\Sigma_2 \arrow[r, "\sigma'_2"'] & \Sigma'
\end{tikzcd}$$

est un diagramme d'amalgation *si pour tout Σ_1-modèle $\mathcal{M}_1 \in |\mathcal{Mod}(\Sigma_1)|$ et tout Σ_2-modèle $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ tels que $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}_2)$, il existe un unique Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ tel que $\mathcal{Mod}(\sigma'_1)(\mathcal{M}') = \mathcal{M}_1$ et $\mathcal{Mod}(\sigma'_2)(\mathcal{M}') = \mathcal{M}_2$.*

$$\begin{tikzcd}
\mathcal{Mod}(\Sigma) & \mathcal{Mod}(\Sigma_1) \arrow[l, "\mathcal{Mod}(\sigma_1)"'] \\
\mathcal{Mod}(\Sigma_2) \arrow[u, "\mathcal{Mod}(\sigma_2)"] & \mathcal{Mod}(\Sigma') \arrow[l, "\mathcal{Mod}(\sigma'_2)"] \arrow[u, "\mathcal{Mod}(\sigma'_1)"']
\end{tikzcd}$$

*Lorqu'on relâche la condition d'unicité sur $\mathcal{M}'$, on parle de diagramme d'*amalgation faible*. Le modèle $\mathcal{M}'$ est appelé* somme amalgamée *de $\mathcal{M}_1$ et $\mathcal{M}_2$.*
On dit que l'institution $\mathcal{I}$ possède la propriété d'amalgation (faible) *pour une classe C de diagrammes commutatifs de morphismes de signatures si et seulement si tout diagramme de C est un diagramme d'amalgation (faible).*

La définition suivante introduit une notion d'équivalence entre modèles qui reflète leur indiscernabilité par les moyens logiques disponibles « à l'intérieur » des institutions, *i.e.* la satisfaction des formules par les modèles.

Définition 1.4.2 (équivalence élémentaire) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution et soit $\Sigma \in |\mathcal{Sig}|$ une signature. Deux Σ-modèles $\mathcal{M}_1, \mathcal{M}_2 \in |\mathcal{Mod}(\Sigma)|$ sont dits* élémentairement équivalents, *noté $\mathcal{M}_1 \equiv_\Sigma \mathcal{M}_2$, si la condition suivante est vérifiée :*

$$\forall \varphi \in \mathcal{Sen}(\Sigma),\ \mathcal{M}_1 \models_\Sigma \varphi \Longleftrightarrow \mathcal{M}_2 \models_\Sigma \varphi$$

De façon équivalente, $\mathcal{M}_1, \mathcal{M}_2$ sont élémentairement équivalents *si $\mathcal{Th}(\mathcal{M}_1) = \mathcal{Th}(\mathcal{M}_2)$.*

Proposition 1.4.1 *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution fermée par négation. Soient $\Sigma \in |\mathcal{Sig}|$ une signature et $T \subseteq \mathcal{Sen}(\Sigma)$ une Σ-théorie complète. Alors, pour tous Σ-modèles $\mathcal{M}, \mathcal{M}' \in |\mathcal{Mod}(T)|$ non-vide, on a $\mathcal{M} \equiv_\Sigma \mathcal{M}'$.*

Preuve (Par l'absurde)
Soit $\varphi \in \mathcal{Sen}(\Sigma)$ telle que $\mathcal{M} \models_\Sigma \varphi$ et $\mathcal{M}' \not\models_\Sigma \varphi$. Puisque $\mathcal{M}' \in \mathcal{Mod}(T)$, φ ne peut appartenir à T par définition. Donc $\varphi \in T$ et, puisque $\mathcal{I}$ est fermée par négation, on a $\mathcal{M}' \models_\Sigma \neg\varphi$. On en déduit que $\mathcal{M}'$ est un modèle de $T \cup \{\neg\varphi\}$ ce qui contredit le fait que T soit une théorie complète, *i.e.* maximale consistante.
□

La notion d'équivalence élémentaire nous permet d'assouplir les égalités entre réduits de modèles dans la définition 1.4.1.

Définition 1.4.3 (amalgation élémentaire (propriété de Robinson))
Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Le diagramme commutatif de morphismes de signatures ci-dessous

$$\begin{tikzcd} \Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma'_1"] \\ \Sigma_2 \arrow[r, "\sigma'_2"'] & \Sigma' \end{tikzcd}$$

est un diagramme d'amalgation élémentaire *si pour tout Σ_1-modèle $\mathcal{M}_1 \in |\mathcal{Mod}(\Sigma_1)|$ et tout Σ_2-modèle $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ tels que*

$$\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \equiv_\Sigma \mathcal{Mod}(\sigma_2)(\mathcal{M}_2)$$

il existe un Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ tel que

$$\mathcal{Mod}(\sigma'_1)(\mathcal{M}') \equiv_{\Sigma_1} \mathcal{M}_1 \text{ et } \mathcal{Mod}(\sigma'_2)(\mathcal{M}') \equiv_{\Sigma_2} \mathcal{M}_2$$

On dira que $\mathcal{I}$ possède la propriété de Robinson *pour une classe C de diagrammes commutatifs de morphismes de signatures si et seulement si tout diagramme de C est un diagramme d'amalgation élémentaire.*

Sous l'hypothèse de la négation, la propriété de Robinson induit aisément le lemme de consistance de Robinson exprimé de la façon suivante.

Proposition 1.4.2 (lemme de consistance de Robinson) *Soit une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ possédant la négation et la propriété de Robinson pour une classe C de diagrammes tels qu'à la définition 1.4.1. Soient $T \subseteq \mathcal{Sen}(\Sigma)$ une Σ-théorie complète et $T_i \supseteq \mathcal{Sen}(\sigma_i)(T)$ ($i \in \{1,2\}$) des Σ_i-théories consistantes. Alors, $T' = \mathcal{Sen}(\sigma_1)(T_1) \cup \mathcal{Sen}(\sigma_2)(T_2)$ est une Σ'-théorie consistante.*

Preuve Puisque T_i ($i \in \{1,2\}$) est une théorie consistante $|\mathcal{Mod}(T_i)| \neq \varnothing$, *i.e.* il existe au moins un Σ_i-modèle $\mathcal{M}_i \in |\mathcal{Mod}(T_i)|$. Du fait de la complétude de T, on a $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \equiv_\Sigma \mathcal{Mod}(\sigma_1)(\mathcal{M}_2)$. Par la propriété de Robinson, il existe alors un Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ tel que $\mathcal{Mod}(\sigma'_i)(\mathcal{M}') \equiv_{\Sigma_i} \mathcal{M}_i$. Par la condition de satisfaction, on a $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma'_i)(T_i)$. On en conclut alors que $\mathcal{Sen}(\sigma_1)(T_1) \cup \mathcal{Sen}(\sigma_2)(T_2)$ possède au moins un modèle et donc qu'elle est consistante. □

Le lemme de consistance de Robinson est un résultat très important pour la théorie de la spécification. Nous verrons en effet au chapitre 2 que pour toute institution fermée par négation il est équivalent à l'interpolation de Craig présentée en section 2.1.1 (*cf.* proposition 2.2.2), cette dernière propriété étant une condition requise (*cf.* section 2.1.2) pour la complétude des systèmes d'inférence structurés présentés en section 1.6 et une certaine forme de modularité des spécifications (*cf.* section 1.7). Malheureusement, le lemme de consistance de Robinson s'appuie sur la notion d'équivalence élémentaire de modèles qui est très difficile à montrer en pratique. Dans la deuxième partie de cette thèse, nous nous intéresserons à la généralisation de la méthode des diagrammes qui est justement une méthode dédiée à cette tâche (*cf.* chapitre 2, partie II).

La propriété d'exactitude, que nous introduisons maintenant, exprime la possibilité d'amalgamer des modèles construits sur des signatures différentes lorsque ceux-ci sont consistants sur une sorte d'« intersection » des signatures (représentée par un pushout).

Définition 1.4.4 (exactitude) *Une institution* $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ *est dite :*

- exacte *si le foncteur* $\mathcal{Mod} : \mathcal{Sig}^{op} \to \mathcal{Cat}$ *préserve les limites finies*[7] *;*
- semi-exacte *si le foncteur* $\mathcal{Mod} : \mathcal{Sig}^{op} \to \mathcal{Cat}$ *préserve les pullbacks*[8] *(i.e. les pushouts de signatures) ;*
- faiblement semi-exacte *si le foncteur* $\mathcal{Mod} : \mathcal{Sig}^{op} \to \mathcal{Cat}$ *préserve les pullbacks faibles*[9]*.*

La semi-exactitude est omniprésente. Toutes les institutions présentant des logiques conventionnelles ou non sont au minimum semi-exactes. En règle générale, les institutions présentant des logiques multi-sortes sont exactes et celles présentant des logiques mono-sortes sont semi-exactes (*cf.* [45]). Dans la plupart des applications cependant, la version faible de la semi-exactitude suffit. Ce point a d'ailleurs fait l'objet de plusieurs travaux (*cf.* [36, 133]) notamment dans les cas multi-paradigmes ou institutions hétérogènes obtenues par une construction de Grothendieck sur une institution (*cf.* [36, 37]).
La propriété d'amalgation est une conséquence directe de la propriété de semi-exactitude comme le montre le corollaire suivant.

Corollaire 1.4.1 *Dans une institution (faiblement) semi-exacte, tout (faible) pushout de morphismes de signatures est un diagramme d'amalgation (faible).*

Preuve Découle directement des définitions. □

1.5 Structuration

Dès que l'on se retrouve confronté à des spécifications de taille réelle, on ne peut pas se contenter de spécifications à plat. Il faut pouvoir se donner les moyens de structurer les spécifications de façon à pouvoir séparer le travail suivant les différentes préoccupations (ou aspects) du système considéré. Pour

7. *cf.* définition en annexe A.
8. *Ibidem.*
9. Dans le sens de propriété universelle faible (*cf.* [78]), *i.e.* n'imposant pas l'unicité.

ce faire, il est nécessaire de définir formellement un *langage de spécification*, *i.e.* un ensemble d'opérations sur les spécifications permettant, à partir des éléments de base donnés par les présentations, de les rassembler pour créer des spécifications plus larges. Dans le cadre des institutions, ceci a donné lieu à la construction inductive des spécifications, ainsi que de leur sémantique, au travers des opérations suivantes.

Définition 1.5.1 (spécification) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution. Pour toute signature $\Sigma \in |\mathcal{S}ig|$, les* Σ-spécifications *et leur* sémantique *sont définies inductivement de la façon suivante :*

1. *pour toute Σ-présentation P, (Σ, P) est une Σ-spécification telle que $\mathcal{M}od((\Sigma, P)) = \mathcal{M}od(P)$;*
2. *pour toutes Σ-spécifications $SP_1 = (\Sigma, \Phi_1)$ et $SP_2 = (\Sigma, \Phi_2)$, $SP_1 \cup SP_2 = (\Sigma, \Phi_1 \cup \Phi_2)$ est une Σ-spécification telle que $\mathcal{M}od(SP_1 \cup SP_2) = \mathcal{M}od(SP_1) \cap \mathcal{M}od(SP_2)$;*
3. *pour tout morphisme de signatures $\sigma \in \mathcal{S}ig(\Sigma_1, \Sigma_2)$, si SP est une Σ_1-spécification, alors $\texttt{translate}\, SP\, \texttt{by}\, \sigma$ est une Σ_2-spécification telle que $\mathcal{M}od(\texttt{translate}\, SP\, \texttt{by}\, \sigma) = \{\mathcal{M} \in \mathcal{M}od(\Sigma_2) / \mathcal{M}od(\sigma)(\mathcal{M}) \in \mathcal{M}od(SP)\}$;*
4. *pour tout morphisme de signatures $\sigma \in \mathcal{S}ig(\Sigma_1, \Sigma_2)$ si SP est une Σ_2-spécification, alors $\texttt{derive from}\, SP\, \texttt{by}\, \sigma$ est une Σ_1-spécification telle que $\mathcal{M}od(\texttt{derive from}\, SP\, \texttt{by}\, \sigma) = \{\mathcal{M}od(\sigma)(\mathcal{M}) / \mathcal{M} \in \mathcal{M}od(SP_2)\}$.*

On note $\mathcal{S}pec_{\mathcal{I}}(\Sigma)$ l'ensemble des Σ-spécifications construites sur l'institution $\mathcal{I}$.

On étend naturellement la relation de conséquence sémantique aux spécifications de la façon suivante.

Définition 1.5.2 (conséquence sémantique) *Soient $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et $\Sigma \in |\mathcal{S}ig|$ une signature. On définit sur $\mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ la* relation de conséquence sémantique, *notée $\models_\Sigma$, de la façon suivante :*
$\forall SP \in \mathcal{S}pec_{\mathcal{I}}(\Sigma), \forall \varphi \in \mathcal{S}en(\Sigma)$,

$$SP \models_\Sigma \varphi \Leftrightarrow \mathcal{M}od(SP) \models_\Sigma \varphi$$

La structuration que nous venons de présenter est souvent qualifiée de *composition horizontale*. Elle permet d'assembler des théories (en fait leurs présentations) pour en faire des spécifications et ainsi construire un système par ajouts successifs de « modules ». Cependant, dans le cadre du processus de développement de logiciels, les spécifications produites à partir du cahier des charges sont souvent trop abstraites pour être exécutables. Le raffinement est alors une méthode de structuration permettant le passage par étapes successives (*i.e.* spécifications intermédiaires) du cahier des charges, donné par une (un ensemble de) spécification(s) « abstraites », au produit final, donné par une (un ensemble de) spécification(s) « exécutable(s) » ou programme qui respecte le comportement décrit par le cahier des charges.

Définition 1.5.3 (raffinement sémantique) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution. Pour toute signature $\Sigma \in |\mathcal{S}ig|$, une Σ-spécification $SP_2 \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ est un* raffinement sémantique *d'une Σ-spécification $SP_1 \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$, noté $SP_1 \rightsquigarrow SP_2$ si :*

$$\mathcal{M}od(SP_2) \subseteq \mathcal{M}od(SP_1)$$

Cette dernière méthode de structuration est souvent qualifiée de *composition verticale* par rapport à la structuration introduite à la définition 1.5.1. Lorsque ces deux méthodes sont utilisées conjointement on parle de composition transversale.

1.6 Système d'inférence

À la différence des institutions, les systèmes d'inférence donnent une vision orientée théorie de la preuve des systèmes logiques qu'ils modélisent. Dans la pratique cette vision est également très importante car elle autorise la définition d'outils d'aide au développement permettant, par exemple, de vérifier mécaniquement (on dita aussi certifier) les preuves (certaines au moins). Intuitivement un système d'inférence consiste en :

- la donnée d'une catégorie de signatures sur lesquelles sont construites les formules ;
- la donnée d'une relation, appelée *relation d'inférence*, modélisant la notion de conséquence.

Nous présentons ici les systèmes d'inférence (*cf.* [90], également appelés π-institution dans [52]) et introduisons les logiques générales (*cf.* [90]) qui réunissent dans un même cadre les notions d'institution et de système d'inférence. L'intérêt d'une telle présentation est qu'elle met en évidence l'importance de l'étude logique-indépendante de propriétés de théorie standard des modèles telles que l'interpolation de Craig par exemple (*cf.* section 2.1.2).

Définition 1.6.1 *Soient $E \in |\mathcal{S}et|$ un ensemble et $\Vdash_E \subseteq \wp(E) \times E$ une relation binaire. La relation $\Vdash_\Sigma$ est :*

réflexive *si pour tout élément $e \in E$, on a $\{e\} \Vdash_E e$;*

monotone *si pour tous sous-ensembles $\Gamma, \Delta \subseteq E$ tels que $\Gamma \subseteq \Delta$ et tout élément $e \in E$ tel que $\Gamma \Vdash_E e$, on a $\Delta \Vdash_E e$;*

transitive *si pour tous sous-ensembles $\Gamma, \Delta \subseteq E$ et tout élément $e \in E$ tels que $\Delta \Vdash_E e$ et $\Gamma \vdash_E \delta$ pour tout $\delta \in \Delta$, on a $\Gamma \Vdash_E e$;*

stable par translation *si pour tout morphisme $\mu \in \mathcal{S}et(E, E')$, tout sous-ensemble $\Gamma \subseteq E$ et tout élément $e \in E$ tels que $\Gamma \Vdash_E e$, on a $\mu(\Gamma) \Vdash_{E'} \mu(e)$.*

Définition 1.6.2 (système d'inférence) *Un* système d'inférence (π-institution) *est la donnée d'un triplet $(\mathcal{S}ig, \mathcal{S}en, \vdash)$ tel que :*

- *$\mathcal{S}ig$ est une catégorie dont les objets sont appelés* signatures *;*
- *$\mathcal{S}en : \mathcal{S}ig \to \mathcal{S}et$ est un* foncteur *qui associe à toute signature $\Sigma \in |\mathcal{S}ig|$ l'ensemble des formules construites au-dessus de Σ ;*
- *$\vdash = (\vdash_\Sigma)_{\Sigma \in |\mathcal{S}ig|}$ est une famille de* relations binaires *telle que pour toute signature $\Sigma \in |\mathcal{S}ig|$, $\vdash_\Sigma \subseteq 2^{\mathcal{S}en(\Sigma)} \times \mathcal{S}en(\Sigma)$ est* réflexive, monotone, transitive *et* stable par translation.

Pour toute signature $\Sigma \in |\mathcal{S}ig|$, la relation $\vdash_\Sigma$ est appelée relation d'inférence *sur $\mathcal{S}en(\Sigma)$.*

Les systèmes d'inférence constituent un cadre abstrait permettant de capturer la notion de conséquence logique indépendamment de la façon dont cette dernière a été définie (calcul, relation de satisfaction, *etc.*). On définit pour les systèmes d'inférence les notions de *présentation*, de *fermeture* d'un ensemble de formules, de *théorie* et de *morphisme de théories* de façon similaire à ce que nous avons fait en section 1.3 en remplaçant simplement $\models$ par $\vdash$ partout où ce symbole apparaît dans les définitions.

Définition 1.6.3 (correction, complétude) *Soient* $\vdash, \vdash' \subseteq \wp(E) \times E$ *deux relations d'inférence sur un même ensemble de formules* E. *On dit que :*

- $\vdash$ *est* correcte *par rapport à* $\vdash'$ *si* $\vdash \subseteq \vdash'$ *;*
- $\vdash$ *est* complète *par rapport à* $\vdash'$ *si* $\vdash' \subseteq \vdash$.

Proposition 1.6.1 *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution. Pour toute signature* $\Sigma \in |\mathcal{S}ig|$, *la relation de conséquence sémantique* $\models_\Sigma \subseteq \wp(\mathcal{S}en(\Sigma)) \times \mathcal{S}en(\Sigma)$ *est réflexive, monotone, transitive et stable par translation, i.e.* $\models_\Sigma$ *est une relation d'inférence sur* $\mathcal{S}en(\Sigma)$.

Preuve Immédiat d'après la définition de $\models_\Sigma$ et la proposition 1.3.2. □

Un exemple typique de l'utilisation des concepts de correction et de complétude est celui où les relations de conséquence sémantique d'une institution donnée sont vues comme des relations d'inférence que l'on cherche à simuler par des techniques orientées preuve. La correction des relations d'inférence orientées-preuve par rapport aux relations de conséquence sémantique (si $\Gamma \vdash_\Sigma \varphi$ alors $\Gamma \models_\Sigma \varphi$) dénote le fait que tout ce que l'on peut prouver syntaxiquement est vrai dans la vision du monde réel donnée par la sémantique alors que la complétude (si $\Gamma \models_\Sigma \varphi$ alors $\Gamma \vdash_\Sigma \varphi$) dénote le fait que l'on peut prouver syntaxiquement tout ce qui est vrai sémantiquement.
Le concept de système d'inférence est aisément définissable au niveau des institutions comme le montre la définition suivante.

Définition 1.6.4 *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution et soit* $\vdash = \{\vdash_\Sigma\}_{\Sigma \in |\mathcal{S}ig|}$ *une famille de relations d'inférence. Le triplet* $(\mathcal{S}ig, \mathcal{S}en, \vdash)$ *est appelé* système d'inférence pour $\mathcal{I}$ *si pour toute signature* $\Sigma \in |\mathcal{S}ig|$, *la relation* $\vdash_\Sigma$ *est une relation d'inférence sur* $\mathcal{S}en(\Sigma)$ *correcte par rapport à la relation* $\models_\Sigma$.

Les logiques générales de J. Meseguer (*cf.* [90]) regroupe les approches théorie des modèles des institutions et théorie de la preuve des systèmes d'inférence dans un cadre unique.

Définition 1.6.5 (logique générale) *Une* logique générale *est la donnée d'un quintuplet* $(\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models, \vdash)$ *tel que :*

- $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *est une institution ;*
- $(\mathcal{S}ig, \mathcal{S}en, \vdash)$ *est un système d'inférence pour* $\mathcal{I}$.

Les systèmes d'inférence que nous venons de définir sont définis en termes de présentations. Ils ne permettent pas de modéliser la notion de conséquence logique d'une spécification structurée. Dans [21], T. Borzyszkowski a défini un ensemble de règles d'inférence associées aux spécifications structurées généralisant les approches proprosées dans [27], [67] et [141]. Il a de plus montré la correction et la complétude de ce système d'inférence structuré pour toute institution fermée par négation et conjonction infinie et possédant l'amalgation faible et l'interpolation de Craig (*cf.* théorème 2.1.3).

Définition 1.6.6 (système d'inférence structuré) *Soit une institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ et soit $I = (\mathcal{S}ig, \mathcal{S}en, \vdash^I)$ le système d'inférence associé à $\mathcal{I}$ (cf. définition 1.6.4). Pour toute signature $\Sigma \in |\mathcal{S}ig|$, la relation d'inférence $\vdash_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ paramétrée par I est définie par l'ensemble suivant de règles (où $\mathfrak{I}$ est un ensemble d'indices) :*

$$(\mathtt{CR})\frac{\{SP \vdash_\Sigma \varphi_i\}_{i\in\mathfrak{I}} \quad \{\varphi_i\}_{i\in I} \vdash^I_\Sigma \varphi}{SP \vdash_\Sigma \varphi} \qquad (\mathtt{basic})\frac{\varphi \in \Gamma}{(\Sigma, \Gamma) \vdash_\Sigma \varphi}$$

$$(\mathtt{sum1})\frac{SP_1 \vdash_\Sigma \varphi}{SP_1 \cup SP_2 \vdash_\Sigma \varphi} \qquad (\mathtt{sum2})\frac{SP_2 \vdash_\Sigma \varphi}{SP_1 \cup SP_2 \vdash_\Sigma \varphi}$$

$$(\mathtt{trans})\frac{SP \vdash_\Sigma \varphi}{\mathtt{translate}\, SP\, \mathtt{by}\, \sigma \vdash_{\Sigma'} \mathcal{S}en(\sigma)(\varphi)} \qquad (\mathtt{derive})\frac{SP' \vdash_{\Sigma'} \mathcal{S}en(\sigma)(\varphi)}{\mathtt{derive\ from}\, SP'\, \mathtt{by}\, \sigma \vdash_\Sigma \varphi}$$

où $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$.

Remarque 1.6.1

1. *Dans [21], le système d'inférence structuré est défini de la même façon que ci-dessus à ceci près que le morphisme de signatures considéré dans la règle* `trans` *est supposé appartenir à une sous-classe distinguée $\mathcal{D} \subseteq \mathcal{S}ig$ de morphismes de signatures et celui de la règle* `derive` *à une autre sous-classe distinguée $\mathcal{T} \subseteq \mathcal{S}ig$ de morphismes de signatures. Dans [21], T. Borzyszkowski suppose que la sous-classe $\mathcal{D}$ est incluse dans la sous-classe $\mathcal{T}$.*
2. *La raison justifiant une telle restriction sur les morphismes de signatures est qu'il existe des logiques pour lesquelles certaines propriétés ne sont vérifiées que sur des diagrammes particuliers. C'est le cas par exemple de l'interpolation de Craig (cf. exemples 2.1.1 et 2.1.2) qui n'est pas vérifiée de façon globale dans la logique du premier ordre multi-sorte mais uniquement sur certains diagrammes de signatures (au moins ceux ne définissant que des injections sur les sortes (cf. [20])). L'intérêt de considérer ces deux sous-classes de morphismes de signatures est de pouvoir identifier ces diagrammes particuliers (cf. section 2.1.2, paragraphe « Structuration » pour une explication plus détaillée).*

L'ensemble de règles ci-dessus définit un *système de preuve compositionnel*, *i.e.* qu'il autorise la preuve de théorèmes pour une spécification SP donnée en fonction de sa structure. Notons que ce système de preuve est paramétré par le système de preuve de la logique sous-jacente.

Notation 1.6.1 *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models^I)$ une institution. Dans la suite, pour toute signature $\Sigma \in |\mathcal{S}ig|$, nous noterons $\vdash^I_\Sigma \subseteq \wp(\mathcal{S}en(\Sigma)) \times \mathcal{S}en(\Sigma)$ la relation d'inférence définie sur les Σ-formules et $\vdash_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ celle définie sur les Σ-spécifications. Nous adoptons la même convention de notation pour les relations de conséquence sémantique, respectivement $\models^I_\Sigma \subseteq \wp(\mathcal{S}en(\Sigma)) \times \mathcal{S}en(\Sigma)$ et $\models_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$.*

Définition 1.6.7 (correction) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution. Pour toute signature $\Sigma \in |\mathcal{S}ig|$, la relation d'inférence $\vdash_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$*

définie ci-dessus est dite correcte *par rapport à la relation de conséquence sémantique* $\models_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ *si pour toute* Σ*-spécification* $SP \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ *et pour toute* Σ*-formule* $\varphi \in \mathcal{S}en(\Sigma)$*, on a :*

$$SP \vdash_\Sigma \varphi \Rightarrow SP \models_\Sigma$$

Définition 1.6.8 (complétude) *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution. Pour toute signature* $\Sigma \in |\mathcal{S}ig|$*, la relation d'inférence* $\vdash_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ *définie ci-dessus est dite* complète *par rapport à la relation de conséquence sémantique* $\models_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}en(\Sigma)$ *si pour toute* Σ*-spécification* $SP \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ *et pour toute* Σ*-formule* $\varphi \in \mathcal{S}en(\Sigma)$*, on a :*

$$SP \models_\Sigma \varphi \Rightarrow SP \vdash_\Sigma \varphi$$

De la même façon que pour la composition horizontale, un ensemble de règles d'inférence a été défini dans [21] caractérisant l'effet de la composition verticale sur les systèmes d'inférence.

Définition 1.6.9 (extension conservative) *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution. Soient* $\Sigma, \Sigma' \in |\mathcal{S}ig|$ *deux signatures,* $SP \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ *et* $SP' \in \mathcal{S}pec_{\mathcal{I}}(\Sigma')$ *deux spécifications sur* $\mathcal{I}$ *et* $\sigma \in \mathcal{S}ig(\Sigma', \Sigma)$ *un morphisme de présentations. On dit que la spécification* SP *est une* extension conservative de SP' suivant σ *si :*

$$\mathcal{M}od(\sigma)(\mathcal{M}od(SP)) = \mathcal{M}od(SP')$$

Définition 1.6.10 (relation de raffinement) *Soit* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *une institution et soit* $(\vdash_\Sigma))_{\Sigma \in |\mathcal{S}ig|}$ *une famille de relations d'inférence sur* $(\mathcal{S}pec_{\mathcal{I}}(\Sigma))_{\Sigma \in |\mathcal{S}ig|}$ *(cf. définition 1.6.6) correcte par rapport à la famille de relations de conséquences sémantiques* $\{\models_\Sigma)\}_{\Sigma \in |\mathcal{S}ig|}$ *(cf. définition 1.5.2). La famille de* relations d'inférence $\rightsquigarrow_\Sigma \subseteq \mathcal{S}pec_{\mathcal{I}}(\Sigma) \times \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ *est définie par l'ensemble suivant de règles :*

$$(\mathtt{basic})\frac{SP \vdash_\Sigma \Gamma}{(\Sigma, \Gamma) \rightsquigarrow_\Sigma SP} \qquad (\mathtt{sum})\frac{SP_1 \rightsquigarrow_\Sigma SP \quad SP_2 \rightsquigarrow_\Sigma SP}{SP_1 \cup SP_2 \rightsquigarrow_\Sigma SP}$$

$$(\mathtt{trans}_1)\frac{SP \rightsquigarrow_\Sigma \mathtt{translate}\, SP' \,\mathtt{by}\, \sigma_1^{-1}}{\mathtt{translate}\, SP \,\mathtt{by}\, \sigma_1 \rightsquigarrow_{\Sigma'} SP'} \qquad (\mathtt{trans}_2)\frac{SP' \rightsquigarrow_{\Sigma'} \mathtt{derive\,from}\, SP'' \,\mathtt{by}\, \sigma_2}{\mathtt{translate}\, SP' \,\mathtt{by}\, \sigma_2 \rightsquigarrow_{\Sigma''} SP''}$$

$$(\mathtt{derive})\frac{SP \rightsquigarrow_{\Sigma''} SP''}{\mathtt{derive\,from}\, SP \,\mathtt{by}\, \sigma_2 \rightsquigarrow_{\Sigma'} SP'}$$

où SP'' *est une extension conservative de* SP *suivant* σ_2

$$(\mathtt{trans-equiv})\frac{\mathtt{translate}\,(\mathtt{translate}\, SP \,\mathtt{by}\, \sigma_1)\,\mathtt{by}\, \sigma_2) \rightsquigarrow_{\Sigma''} SP''}{\mathtt{translate}\, SP \,\mathtt{by}\, \sigma_2 \circ \sigma_1 \rightsquigarrow_{\Sigma''} SP''}$$

où $\sigma_1 \in \mathcal{S}ig(\Sigma, \Sigma')$ *et* $\sigma_2 \in \mathcal{S}ig(\Sigma', \Sigma'')$.

La définition ci-dessus s'inspire largement de celles données dans [142, 27]. La précision apportée sur la nature de SP'' dans la règle `derive` ne peut être modélisée, et donc vérifiée de façon interne. Néanmoins, il apparaît dans [142, 27] que lorsque σ est une inclusion, SP'' est, par définition, une extension conservative de SP'. Une vision plus « théorie de modèles » de cette condition peut-être trouvée dans [51].

Remarque 1.6.2 *De même que précedemment, les relations de raffinement dans [21] sont définies de la même façon que ci-dessus à ceci près que le morphisme σ_1 dans les règles* $\texttt{trans}_1$ *et* `trans-equiv` *est supposé être un isomorphisme et le morphisme σ_2 dans les règles* $\texttt{trans}_2$, `derive` *et* `trans-equiv` *est supposé appartenir à la sous-classe $\mathcal{D}$. T. Borzyszkowski suppose de plus la condition suivante : $\mathcal{T} = \mathcal{D} \circ Iso$, i.e. la sous-classe $\mathcal{T}$ est égale à la sous-classe $\mathcal{D}$ à isomorphisme près.*

1.7 Modularité

Le terme de modularité s'applique de façon générale à ce qui est constituté de « modules », *i.e.* d'éléments aptes à toutes sortes de combinaisons conservatives. On entend par combinaison conservative le fait que le comportement global est équivalent à la « somme » des comportements respectifs de chacune des parties. Dans le cadre des systèmes informatiques, ceci se traduit principalement par la composabilité des étapes de raffinement et la correction des spécifications paramétrées. Depuis les travaux de D. Parnas (*cf.* [105]), il est universellement reconnu que la modularité des systèmes informatiques est indispensable pour le développement de logiciels de qualité. La propriété de modularisation est un outil permettant d'assurer cette nécessaire modularité dont l'importance (de certaines versions) pour le développement formel de programmes a été soulignée dans différents travaux (*cf.* [13, 48, 50, 83, 111]). Dans le cadre particulier de la théorie de la preuve, différents travaux (*cf.* [138, 140, 47, 46, 137, 139, 82]) ont amené une généralisation de la propriété de modularisation et permis de découvrir et d'étudier les liens de cette dernière avec (différentes versions de) l'interpolation de Craig. Nous définissons ici la propriété de modularisation et montrons son importance pour le développement formel pas-à-pas de logiciels. Nous renvoyons le lecteur à la section 2.1.2 pour un exposé des liens entre la propriété de modularisation et une variante de l'interpolation de Craig appelée interpolation de Craig-Robinson.

Implémentation de spécifications ([50, 83])

La notion de raffinement définie dans la section précédente se fait à signature constante. Dans la pratique, même s'il est toujours possile de se ramener à ce cas, il est plus commode de considérer un changement de signatures lors du raffinement. Ceci permet de renommer et de rajouter des symboles pour se rapprocher de plus en plus de la syntaxe d'un langage de programmation. On parle alors d'implémentation.

Considérons donc une spécification $SP = (\Sigma, \Phi)$ que l'on souhaite implémenter sur une spécification $SP' = (\Sigma', \Phi')$. Pour ce faire, il faut tout d'abord considérer une spécification $SP'' = (\Sigma'', \Phi'')$ telle qu'il existe un morphisme de présentations $\sigma : \Phi' \to \Phi''$. Puisqu'on désire que la spécification SP'' soit une image « fidèle » de la spécification SP', l'ensemble Φ'' ne doit pas imposer de contraintes non-spécifiées dans Φ'. On demande alors naturellement que la spécification SP'' soit une extension conservative de la spécification SP' suivant σ. Il est maintenant nécessaire de mettre en relation les deux spécifications SP

$$\begin{tikzcd} SP \arrow[r, "\sigma'"] & SP'' \\ & SP' \arrow[u, "\sigma"'] \end{tikzcd}$$

FIG. 1.1 – *Étape de raffinement*

et SP''. Seulement, on veut pouvoir préserver le comportement de SP tel que décrit par ses axiomes. Il suffit pour cela qu'il existe un morphisme de présentations $\sigma' : \Phi \to \Phi''$, *i.e.* toute conséquence de SP est une conséquence de SP''. On dit alors que SP'' interprète SP.
Au final, une ***implémentation*** de $SP = (\Sigma, \Phi)$ sur $SP' = (\Sigma', \Phi')$ est donc la donnée d'un morphisme de présentations $\sigma' : \Phi \to \Phi''$ où $SP'' = (\Sigma, \Phi'')$ est une extension conservative (usuellement appelée ***spécification médiane***) de SP'. Un « triangle d'implémentation » tel que celui de la figure 1.1 est parfois appelé « étape canonique de raffinement » (*cf.* [136]). La difficulté réside ici dans le fait que la spécification médiane SP'' doit être à la fois suffisamment riche pour interpréter SP et suffisamment pauvre pour être une extension conservative de SP'.

S'il est intéressant de pouvoir raffiner des spécifications, il est encore plus intéressant de pouvoir composer les étapes de raffinement. Ceci permet alors de développer des logiciels de façon abstraite en ne s'intéressant qu'à ce que le système est censé faire (le quoi) et de se rapprocher petit à petit, *i.e.* par raffinements successifs, d'une spécification exécutable, ou programme, décrivant comment le système exécute ses actions (le comment). Le problème est alors le suivant. Considérons deux étapes consécutives de raffinement (*cf.* figure 1.1) : la première de SP_1 sur SP_2 (avec SP' la spécification médiane) et la seconde de SP_2 sur SP_3 (avec SP'' la spécification médiane). On souhaite donc composer ces deux implémentations de façon simple et naturelle de façon à obtenir une implémentation de SP_1 directement sur SP_3. La question qui se pose alors est : que peut être la spécification médiane ? C'est là que la propriété de modularisation entre en jeu. Elle permet en effet de compléter le carré dont σ_1 et σ'_2 forment deux des quatre côtés en fournissant :
- une spécification SP''' ;
- un morphisme de présentations $\sigma'_3 : SP' \to SP'''$;
- une extension conservative $\sigma_3 : SP'' \to SP'''$.

On obtient bien alors une implémentation directe de SP_1 sur SP_3 dont la spécification médiane est SP'''.

Spécifications paramétrées

Le fait de voir une spécification comme la donnée d'un « contexte » et d'un paramètre s'est révélé particulièrement utile dans la pratique. Ceci permet en

$$
\begin{tikzcd}
SP_1 \arrow[r, "\sigma'_1"] & SP' \arrow[r, dashed, "\sigma'_3"] & SP''' \\
& SP_2 \arrow[u, "\sigma_1"'] \arrow[r, "\sigma'_2"] & SP'' \arrow[u, dashed, "\sigma_3"'] \\
& & SP_3 \arrow[u, "\sigma_2"']
\end{tikzcd}
$$

FIG. 1.2 – *Composition d'étapes de raffinement*

effet d'instancier un même contexte dans différentes situations suivant le choix de valeurs pour les paramètres. De telles spécifications sont communément appelées *spécifications paramétrées* (*cf.* [48, 50]).
Une spécification paramétrée définit donc une « famille » de spécifications partageant un même cadre. On peut ainsi imaginer une spécification paramétrée `Liste[Données]` définissant des opérations sur des listes (ajout d'un élément, suppression d'un élément, concaténation de deux listes, *etc.*), indépendamment de la nature des éléments contenus dans la liste (le paramètre `Données`). La spécification des listes d'entiers, par exemple, est obtenue à partir de la spécification `Liste[Données]` en instanciant le paramètre `Données` par la spécification des entiers naturels. De ce fait, certaines approches tendent à voir les spécifications paramétrées comme des fonctions sur les modèles ou bien sur les spécifications. Il est néanmoins parfois utile de les voir aussi comme des spécifications « normales ».

Une spécification S est dite *paramétrée* par une spécification SP_1 (appelée *paramètre*) si S est une extension conservative de SP_1. On note alors $S[SP_1]$ plutôt que SP. Une *instanciation de paramètre* est un morphisme de théories σ' : $SP_1 \to SP_2$. La situation est alors celle décrite par le diagramme de la figure 1.3 où $S[SP_1]$ est la spécification paramétrée (par exemple `Liste[Données]`), SP_1 est la spécification du paramètre (par exemple `Données`), SP_2 est la spécification d'une instance de SP_1 (par exemple la spécification `Nat` des entiers naturels) et $S[SP_2]$ est la spécification paramétrée dans laquelle le paramètre a été instancié (on alors la spécification `Liste[Nat]` des listes d'entiers naturels).

$$
\begin{tikzcd}
S[SP_1] \arrow[r, dashed] & S[SP_2] \\
SP_1 \arrow[u, "\sigma"] \arrow[r, "\sigma'"'] & SP_2 \arrow[u, dashed]
\end{tikzcd}
$$

FIG. 1.3 – *Spécification paramétrée et instanciation*

La propriété de modularisation nous permet ici encore de compléter le carré dont σ et σ' forment deux des quatre côtés. La spécification $S[SP_2]$ ainsi obtenue est alors bien une extension conservative de l'argument SP_2.

Propriété de modularisation

Nous présentons maintenant la propriété de modularisation telle que donnée dans [47]. Cette version de la propriété de modularisation généralise les versions précédentes (*cf.* [82, 136, 139]) en la localisant notamment sur un pushout de théories.
Soient donc trois spécifications SP_1, SP_2 et SP_3 telles que SP_3 est consistante. Soient $\sigma : SP_1 \to SP_2$ une extension conservative et $\sigma' : SP_1 \to SP_3$ un morphisme de présentations (*cf.* figure 1.4). Ainsi que nous l'avons indiqué dans les deux sections précédentes, ceci est la situation dans laquelle on se trouve lorsqu'on veut composer des étapes de raffinement ou lorsqu'on instancie une spécification paramétrée.

Définition 1.7.1 (propriété de modularisation) *Un système d'inférence* $I = (\mathcal{S}ig, \mathcal{S}en, \vdash)$ *possède la* propriété de modularisation *si le pushout*[10] *de tout extension conservative* $\sigma : SP_1 \to SP_2$ *le long de tout morphisme* $\sigma' : SP_1 \to SP_3$ *(cf. figure 1.4) est également une extension conservative, i.e.* $\sigma'' : SP_3 \to SP$ *est une extension conservative.*

$$\begin{tikzcd} SP_2 \arrow[r, dashed] & SP \\ SP_1 \arrow[u, "\sigma"] \arrow[r, "\sigma'"'] & SP_3 \arrow[u, dashed, "\sigma''"'] \end{tikzcd}$$

FIG. 1.4 – *Pushout d'extension conservative*

La propriété de modularisation permet donc de compléter les carrés des figures 1.2 et 1.3 en construisant une spécification SP satisfaisant les conditions énoncées dans les deux sections précédentes, *i.e.* elle étend SP_3 de façon conservative et interprète SP_2. On parle alors de stabilité des extensions conservatives par pushout.

1.8 Flèches entre institutions

Ainsi que nous l'indiquions en introduction, travailler avec un système logique arbitraire mais fixe n'est pas toujours aisé. L'utilisation de différents systèmes logiques est, en effet, parfois nécessaire pour la spécification et le développement d'un même logiciel. Différentes étapes de développement (analyse des besoins, spécification, implémentation, *etc.*), différents aspects (contrôle, flux de donnée, *etc.*), différents modules issus de systèmes hétérogènes (mélange de composants hardware et software, ces derniers pouvant ne pas tous répondre du même paradigme de programmation), différents niveaux de formalisation et

10. *cf.* définition en annexe A.

de détails, différents outils ou notations appellent l'utilisation de différents systèmes logiques. Le problème qui se pose alors peut être formulé de la façon suivante :

Est-il possible de réutiliser, pour un système logique donné, des spécifications, outils, librairies et stratégies de preuves définis pour un autre système logique ?

La résolution de ce type de problème dépasse de loin la définition de mécanismes de structuration des spécifications (*cf.* [19, 116]) ou la modélisation du processus de développement (*cf.* [117]) dans un système logique fixe. Il est donc nécessaire de fournir des bases théoriques solides permettant cette résolution. Pour ce faire, les systèmes logiques concernés doivent être, d'une façon ou d'une autre, mis en relation les uns avec les autres.

Utilisant le langage de la théorie des catégories, les relations entre systèmes logiques, lorsque ceux-ci sont présentés par des institutions, sont naturellement représentées par une notion de flèche entre les institutions concernées. Différentes notions de flèches entre institutions ont été proposées, à commencer par les morphismes d'institutions (*cf.* [132, 28, 86] pour un état de l'art et une étude détaillée). Ces différentes notions de flèches peuvent être classées en deux familles en fonction de l'utilisation qu'on en fait, à savoir :

1. construire une institution riche $\mathcal{I}'$ au-dessus d'une institution $\mathcal{I}$ plus pauvre (*cf.* définition 1.8.1);
2. représenter une institution $\mathcal{I}$ dans une institution $\mathcal{I}'$ suffisamment riche pour cela (*cf.* définition 1.8.2).

Alors que la première de ces familles de flèches se caractérise par le transport covariant des signatures et des modèles, ce transport est contravariant pour les flèches de la deuxième famille.

Dans [86], A. Martini et U. Wolter étudient de façon systématique les principales notions de flèche entre institutions au travers du cadre théorique des « power institutions » : morphismes, simulations, transformations et applications (pleines et simples). Ils montrent en particulier que ces différentes notions de flèches englobent un certain nombre de constructions élémentaires fondamentales telles que la réduction du typage le long des morphismes de signatures, l'emprunt de modèles, le partage de la théorie des modèles et les restrictions sémantiques.

Nous restreignons notre présentation aux deux notions principales de flèches entre institutions, *i.e.* les morphismes et les co-morphismes d'institutions, les autres notions de flèches entre institutions n'étant que des variantes définies dans un cadre précis pour la résolution d'un problème particulier. Pour un état de l'art plus détaillé des différentes notions de flèches entre institutions et une étude détaillée des relations entre ces différentes notions, nous renvoyons le lecteur intéressé à la thèse de M. Cerioli (*cf.* [28]), à l'article de A. Tarlecki (*cf.* [132]) ainsi qu'à l'article de A. Martini et U. Wolter (*cf.* [86]).

Morphismes Un morphisme d'institutions exprime le fait qu'une institution « riche » est construite au-dessus d'une institution « pauvre ». Intuitivement, un morphisme d'institutions $\mu : \mathcal{I}' \to \mathcal{I}$ consiste en la donnée (ce qui suit donne l'intuition de ce qui se passe dans les cas typiques de morphismes mais pas nécessairement dans tous les cas):

– une translation des $\mathcal{I}'$-signatures vers les $\mathcal{I}$-signatures. L'idée ici est qu'une

$\mathcal{I}'$-signature Σ est projetée sur une $\mathcal{I}$-signature $\Phi(\Sigma)$ en laissant de côté toutes les particularités de $\mathcal{I}'$ n'ayant pas de contrepartie dans $\mathcal{I}$. Par exemple, si les signatures de $\mathcal{I}'$ possèdent des symboles de fonctions totales et partielles alors que celles de $\mathcal{I}$ ne possèdent que des symboles de fonctions totales, la projection retirera simplement les symboles de fonctions partielles des signatures de $\mathcal{I}'$;

- une translation des $\mathcal{I}$-formules vers les $\mathcal{I}'$-formules. Notons que l'inverse n'est pas envisageable. En effet, si on voit les Σ-formules comme des arbres de dérivation décorés avec des symboles de Σ, il n'est alors pas évident de savoir que faire de ces symboles qui sont écartés durant la projection. Il est par contre plus simple de translater les $\Phi(\Sigma)$-formules vers les Σ-formules puisque les premières forment un sous-ensemble des dernières. Intuitivement, on garde les formules telles qu'elles sont ;
- une translation des $\mathcal{I}'$-modèles vers les $\mathcal{I}$-modèles. Intuitivement, ceci consiste en une translation d'un Σ-modèle vers un $\Phi(\Sigma)$-modèle en oubliant l'interprétation sémantique de tous les symboles laissés de côté par la translation de Σ vers $\Phi(\Sigma)$.

De plus, on impose sur les morphismes d'institutions une condition comparable à la condition de satisfaction. Intuitivement, cette condition requiert le fait qu'un modèle translaté satisfait une formule si et seulement si le modèle original satisfait la formule translatée.

Définition 1.8.1 (morphisme) *Soient deux institutions* $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ *et* $\mathcal{I}' = (\mathcal{S}ig', \mathcal{S}en', \mathcal{M}od', \models')$. *Un* morphisme d'institutions $\mu : \mathcal{I}' \to \mathcal{I}$ *est la donnée d'un triplet* (Φ, α, β) *tel que :*

- $\Phi : \mathcal{S}ig' \to \mathcal{S}ig$ *est un* foncteur *;*
- $\alpha : \mathcal{S}en \circ \Phi \Rightarrow \mathcal{S}en'$ *est une* transformation naturelle, *i.e. une famille de foncteurs indexée par* $|\mathcal{S}ig'|$ *telle que pour tout morphisme de signatures* $\sigma' \in \mathcal{S}ig'(\Sigma'_1, \Sigma'_2)$ *le diagramme de la figure 1.5 commute.*

$$\begin{tikzcd}
\Sigma'_2 & \mathcal{S}en(\Phi(\Sigma'_2)) \arrow[r, "\alpha_{\Sigma'_2}"] & \mathcal{S}en'(\Sigma'_2) \\
\Sigma'_1 \arrow[u, "\sigma'"] & \mathcal{S}en(\Phi(\Sigma'_1)) \arrow[u, "\mathcal{S}en(\Phi(\sigma'))"] \arrow[r, "\alpha_{\Sigma'_1}"'] & \mathcal{S}en'(\Sigma'_1) \arrow[u, "\mathcal{S}en'(\sigma')"']
\end{tikzcd}$$

FIG. 1.5 – *Translation de formules*

- $\beta : \mathcal{M}od' \Rightarrow \mathcal{M}od \circ \Phi^{op}$ *est une* transformation naturelle, *i.e. une famille de foncteurs indexée par* $|\mathcal{S}ig'|$ *telle que pour tout morphisme de signatures* $\sigma' \in \mathcal{S}ig'(\Sigma'_1, \Sigma'_2)$ *le diagramme de la figure 1.6 commute.*

On impose de plus la condition suivante :
$\forall \Sigma' \in |\mathcal{S}ig'|, \forall \varphi \in \mathcal{S}en(\Phi(\Sigma')), \forall \mathcal{M}' \in |\mathcal{M}od'(\Sigma')|,$

$$\mathcal{M}' \models'_{\Sigma'} \alpha_{\Sigma'}(\varphi) \Leftrightarrow \beta_{\Sigma'}(\mathcal{M}') \models_{\Phi(\Sigma')} \varphi$$

Exemple 1.8.1 *Un exemple typique de morphisme d'institutions est celui existant entre la logique du premier ordre (FOL) et la logique équationnelle (EQL)*

$$\begin{tikzcd}
\Sigma'_2 & \mathcal{M}od'(\Sigma'_2) \arrow[r, "\beta_{\Sigma'_2}"] \arrow[d, "\mathcal{M}od'(\sigma')"'] & \mathcal{M}od(\Phi(\Sigma'_2)) \arrow[d, "\mathcal{M}od(\Phi(\sigma'))"] \\
\Sigma'_1 \arrow[u, "\sigma'"] & \mathcal{M}od'(\Sigma'_1) \arrow[r, "\beta_{\Sigma'_1}"'] & \mathcal{M}od(\Phi(\Sigma'_1))
\end{tikzcd}$$

FIG. 1.6 – *Translation de modèles*

(cf. exemples 1.1.3 et 1.1.6). Le foncteur sur les signatures oublie les symboles de relations, i.e. il associe à toute signature du premier ordre $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ la signature algébrique $(\mathcal{S}, \mathcal{F})$. Pour ce qui est des formules, chaque $(\mathcal{S}, \mathcal{F})$-équation peut être vue comme une $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-formule ; ceci définit une famille de foncteurs de translation entre les ensembles de formules. Enfin, chaque $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-modèle peut être vu comme une $(\mathcal{S}, \mathcal{F})$-algèbre en oubliant l'interprétation des symboles de relation ; ceci définit une famille de foncteurs de translation entre les catégories de modèles.

Les morphismes d'institutions permettent de comparer de façon rudimentaire des institutions entre elles. Puisqu'ils transportent à la fois les modèles et les formules, ils permettent également le transport de spécifications d'une institution dans une autre au travers des deux opérateurs suivants (où $\mu : \mathcal{I}' \to \mathcal{I}$ est un morphisme d'institutions) :

dérivation : si $SP' \in \mathcal{S}pec_{\mathcal{I}'}(\Sigma')$ est une $\mathcal{I}'$-spécification, alors $\texttt{derive from}\, SP' \,\texttt{via}\, \mu \in \mathcal{S}pec_{\mathcal{I}}(\Phi(\Sigma'))$ est une spécification sur $\mathcal{I}$ telle que $\mathcal{M}od(\texttt{derive from}\, SP' \,\texttt{via}\, \mu) = \{\beta_{\Sigma'}(\mathcal{M}')/\mathcal{M}' \in \mathcal{M}od(SP')\}$;

translation : si $SP \in \mathcal{S}pec_{\mathcal{I}}(\Sigma)$ est une $\mathcal{I}$-spécification, alors $\texttt{translate}\, SP \,\texttt{via}\, \mu \,\texttt{to}\, \Sigma' \in \mathcal{S}pec_{\mathcal{I}'}(\Sigma')$ est une $\mathcal{I}'$-spécification telle que $\mathcal{M}od(\texttt{translate}\, SP \,\texttt{via}\, \mu \,\texttt{to}\, \Sigma') = \{\mathcal{M}' \in |\mathcal{M}od'(\Sigma')|/\beta_{\Sigma'}(\mathcal{M}') \in \mathcal{M}od(SP)\}$.

Les morphismes d'institutions permettent également la réutilisation d'outils de preuve d'une institution à une autre. On a, en effet, les résultats suivants.

Proposition 1.8.1 ([130]) *La relation de conséquence sémantique est préservée par les morphismes d'institutions : pour tout morphisme d'institutions $\mu : \mathcal{I}' \to \mathcal{I}$, toutes signatures $\Sigma \in |\mathcal{S}ig|$ et $\Sigma' \in |\mathcal{S}ig'|$ telles que $\Phi(\Sigma') = \Sigma$, tout ensemble de Σ-formules $\Psi \subseteq \mathcal{S}en(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, si $\Psi \models_\Sigma \varphi$ alors $\alpha_{\Sigma'}(\Psi) \models'_{\Sigma'} \alpha_{\Sigma'}(\varphi)$.*

En règle générale, la relation de conséquence sémantique n'est pas reflétée par les morphismes d'institutions, *i.e.* il n'est pas possible d'obtenir le sens inverse de l'implication de la proposition précédente. Pour transformer cette implication en équivalence, d'autres hypothèses doivent être faites.

Théorème 1.8.1 ([132]) *Pour tout morphisme d'institutions $\mu : \mathcal{I}' \to \mathcal{I}$, toutes signatures $\Sigma \in |\mathcal{S}ig|$ et $\Sigma' \in |\mathcal{S}ig'|$ telles que $\Phi(\Sigma') = \Sigma$, tout ensemble de Σ-formules $\Psi \subseteq \mathcal{S}en(\Sigma)$ tel que pour tout modèle $\mathcal{M} \in \mathcal{M}od(\Psi)$ il existe un Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ tel que $\beta_{\Sigma'}(\mathcal{M}') = \mathcal{M}$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, on a $\Psi \models_\Sigma \varphi$ si et seulement si $\alpha_{\Sigma'}(\Psi) \models'_{\Sigma'} \alpha_{\Sigma'}(\varphi)$.*

Corollaire 1.8.1 ([132]) *La relation de conséquence sémantique est préservée et reflétée par les morphismes d'institutions induisant une translation surjective sur les modèles : pour tout morphisme d'institutions $\mu : \mathcal{I}' \to \mathcal{I}$, toutes signatures $\Sigma \in |\mathcal{S}ig|$ et $\Sigma' \in |\mathcal{S}ig'|$ telles que $\Phi(\Sigma') = \Sigma$ et, pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$, il existe un Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ tel que $\beta_{\Sigma'}(\mathcal{M}') = \mathcal{M}$, tout ensemble de Σ-formules $\Psi \subseteq \mathcal{S}en(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, on a $\Psi \models_\Sigma \varphi$ si et seulement si $\alpha_{\Sigma'}(\Psi) \models'_{\Sigma'} \alpha_{\Sigma'}(\varphi)$.*

Comorphismes De façon duale par rapport aux morphismes, les comorphismes [11] d'institutions expriment le fait qu'une institution $\mathcal{I}$ est incluse dans une institution $\mathcal{I}'$. Intuitivement, un comorphisme d'institutions consiste en la donnée :

- une translation Φ des $\mathcal{I}$-signatures vers les $\mathcal{I}'$-signatures. Étant donnée une $\mathcal{I}$-signature Σ, il s'agit alors de trouver un encodage adéquat $\Phi(\Sigma)$ de Σ. En particulier, la catégorie de modèles associée à $\Phi(\Sigma)$ doit approximer la catégorie de modèles associée à Σ ;
- une translation α des $\mathcal{I}$-formules vers les $\mathcal{I}'$-formules. La raison pour laquelle la translation des formules se fait dans le même sens que la translation des signatures est similaire à la raison pour laquelle la translation des formules au sein d'une institution se fait dans le même sens que les morphismes de signatures. En effet, si une $\mathcal{I}$-signature Σ est encodée dans une $\mathcal{I}'$-signature $\Phi(\Sigma)$, on s'attend à ce que chaque symbole de Σ soit translaté vers un symbole correspondant dans $\Phi(\Sigma)$. Maintenant, si on considère une Σ-formule φ comme étant un arbre de dérivation décoré avec des symboles de Σ, la translation α_Σ conserve alors la structure de l'arbre et transforme simplement les symboles de l'arbre suivant la translation $\Phi(\Sigma)$;
- une translation β des $\mathcal{I}'$-modèles vers les $\mathcal{I}$-modèles donnant la relation entre les Σ-modèles de $\mathcal{I}$ et les $\Phi(\Sigma)$-modèles de $\mathcal{I}'$. De même que dans la définition des institutions, on a ici une contravariance de la translation de modèles. Il arrive souvent qu'on ait également une translation de modèles γ dans la direction opposée. Cependant, alors que β définit une transformation naturelle, ce n'est pas toujours le cas pour γ (*cf.* [73] pour un contre-exemple). D'autre part, la naturalité de β est essentielle pour les spécifications hétérogènes (*cf.* [96]).

De plus, on impose sur les comorphismes d'institutions une condition de satisfaction similaire à celle imposée sur les morphismes d'institutions.

Définition 1.8.2 (comorphisme) *Soient $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ et $\mathcal{I}' = (\mathcal{S}ig', \mathcal{S}en', \mathcal{M}od', \models')$ deux institutions. Un* comorphisme d'institutions *$\mu : \mathcal{I} \to \mathcal{I}'$ est la donnée d'un triplet (Φ, α, β) tel que :*

- *$\Phi : \mathcal{S}ig \to \mathcal{S}ig'$ est un* foncteur *;*
- *$\alpha : \mathcal{S}en \Rightarrow \mathcal{S}en' \circ \Phi$ est une* transformation naturelle*, i.e. une famille de foncteurs indexée par $|\mathcal{S}ig'|$ telle que pour tout morphisme de signatures $\sigma' \in \mathcal{S}ig'(\Sigma'_1, \Sigma'_2)$ le diagramme de la figure 1.7 commute.*

11. Le terme de comorphisme n'est pas rigoureusement correct. En effet, les comorphismes d'institutions ne sont duaux par rapport aux morphismes d'institutions que sur les composantes signatures et modèles. Nous employons ici le terme générique de comorphisme pour désigner toutes flèches entre institutions duales par rapport aux morphismes sur ces deux composantes. Des exemples de comorphismes dans la littérature sont les simulations (*cf.* [28]), les représentations (*cf.* [132]) ou encore les applications pleines et simples (*cf.* [90]).

$$\begin{tikzcd}
\Sigma'_2 & \mathcal{S}en(\Sigma_2) \arrow[r, "\alpha_{\Sigma_2}"] & \mathcal{S}en(\Phi(\Sigma_2)) \\
\Sigma'_1 \arrow[u, "\sigma'"] & \mathcal{S}en(\Sigma'_1) \arrow[u, "\mathcal{S}en(\sigma)"] \arrow[r, "\alpha_{\Sigma_1}"'] & \mathcal{S}en(\Phi(\Sigma_1)) \arrow[u, "\mathcal{S}en(\Phi(\sigma))"']
\end{tikzcd}$$

FIG. 1.7 – *Translation de formules*

– $\beta : \mathcal{M}od' \circ \Phi^{op} \Rightarrow \mathcal{M}od$ *est une* transformation naturelle*, i.e. une famille de foncteurs indexée par $\mathcal{S}ig'$ telle que pour tout morphisme de signatures $\sigma' \in \mathcal{S}ig'(\Sigma'_1, \Sigma'_2)$ le diagramme de la figure 1.8 commute.*

$$\begin{tikzcd}
\Sigma'_2 & \mathcal{M}od'(\Phi(\Sigma_2)) \arrow[r, "\beta_{\Sigma_2}"] \arrow[d, "\mathcal{M}od'(\Phi(\sigma))"'] & \mathcal{M}od(\Sigma_2) \arrow[d, "\mathcal{M}od(\sigma)"] \\
\Sigma'_1 \arrow[u, "\sigma'"] & \mathcal{M}od'(\Phi(\Sigma_1)) \arrow[r, "\beta_{\Sigma_1}"'] & \mathcal{M}od(\Sigma_1)
\end{tikzcd}$$

FIG. 1.8 – *Translation de modèles*

On impose de plus la condition suivante :
$\forall\Sigma \in |\mathcal{S}ig|, \forall\varphi \in \mathcal{S}en(\Sigma), \forall\mathcal{M}' \in |\mathcal{M}od'(\Phi(\Sigma))|,$

$$\mathcal{M}' \models'_{\Phi(\Sigma)} \alpha_\Sigma(\varphi) \Leftrightarrow \beta_\Sigma(\mathcal{M}') \models_\Sigma \varphi$$

Exemple 1.8.2 *Un exemple typique de comorphisme d'institutions est celui existant entre la logique équationnelle (EQL) et la logique du premier ordre (FOL) avec égalité (le dual du morphisme défini dans l'exemple 1.8.1). Une signature algébrique est translatée vers une signature du premier ordre dont l'ensemble de symboles de relations est vide. La translation des formules est simplement l'inclusion des équations dans l'ensemble des formules du premier ordre. Enfin, un modèle du premier ordre est translaté en ne considérant que l'algèbre sous-jacente, i.e. en oubliant l'interprétation des symboles de relations.*

Les comorphismes d'institutions permettent également la réutilisation d'outils de preuve entre institutions. On a en effet les résultats suivant.

Proposition 1.8.2 ([132]) *La relation de conséquence sémantique est préservée par les comorphismes d'institutions : pour tout comorphisme d'institutions $\rho : \mathcal{I} \to \mathcal{I}'$, toute signature $\Sigma \in |\mathcal{S}ig|$, tout ensemble de Σ-formules $\Psi \subseteq \mathcal{S}en(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, si $\Psi \models_\Sigma \varphi$ alors $\alpha_\Sigma(\Psi) \models'_{\Phi(Sigma)} \alpha_\Sigma(\varphi)$.*

Théorème 1.8.2 ([132]) *Pour tout comorphisme d'institutions $\rho : \mathcal{I} \to \mathcal{I}'$, toute signature $\Sigma \in |\mathcal{S}ig|$, tout ensemble de Σ-formules $\Psi \subseteq \mathcal{S}en(\Sigma)$ tel que pour tout modèle $\mathcal{M} \in \mathcal{M}od(\Psi)$ il existe un Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Phi(\Sigma))|$ tel que $\beta_\Sigma(\mathcal{M}') = \mathcal{M}$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, on a $\Psi \models_\Sigma \varphi$ si et seulement si $\alpha_\Sigma(\Psi) \models'_{\Phi(\Sigma)} \alpha_\Sigma(\varphi)$.*

Corollaire 1.8.2 ([132]) *La relation de conséquence sémantique est préservée et reflétée par les comorphismes d'institutions induisant une translation surjective sur les modèles : pour toute comorphisme d'institutions* $\rho : \mathcal{I} \to \mathcal{I}'$*, toute signature* $\Sigma \in |\mathcal{S}ig|$ *telle que pour tout* Σ*-modèle* $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$*, il existe un* Σ'*-modèle* $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ *tel que* $\beta_\Sigma(\mathcal{M}') = \mathcal{M}$*, tout ensemble de* Σ*-formules* $\Psi \subseteq \mathcal{S}en(\Sigma)$ *et toute* Σ*-formule* $\varphi \in \mathcal{S}en(\Sigma)$*, on a* $\Psi \models_\Sigma \varphi$ *si et seulement si* $\alpha_\Sigma(\Psi) \models'_{\Phi(\Sigma)} \alpha_\Sigma(\varphi)$.

Ce sont exactement les mêmes résultats que ceux obtenus pour les morphismes d'institutions. Seulement cette fois, la façon dont on quantifie sur les signatures est plus naturelle.

Chapitre 2

Interpolation de Craig

2.1 Introduction

La structuration et la modularité présentées dans le chapitre précédent sont des propriétés importantes pour le développement de logiciels de qualité. De façon à s'abstraire des contingences dues à un système logique particulier, une partie des travaux qui leur ont été consacrés ont été réalisés dans le cadre des institutions (structuration *cf.* [18, 27, 51, 116, 115, 130, 142]) et des logiques générales (modularité *cf.* [138, 140, 47, 46, 137]). En ce qui concerne la structuration, outre la définition d'un ensemble d'opérateurs sur les spécifications permettant de les structurer, ces travaux se sont intéressé à l'extension aux spécifications structurées des systèmes d'inférence associés au système logique sous-jacent. La complétude de ces systèmes d'inférence a également été étudiée, notamment dans [27] et [18] qui ont montré que l'interpolation de Craig est une condition requise pour assurer cette complétude. Pour ce qui est de la modularité, l'étude du théorème de modularisation dans le cadre des logiques générales a permis de mettre en avant le caractère suffisant de l'interpolation de Craig pour la validité de ce théorème.

Étant donnée l'importance de la propriété d'interpolation de Craig pour la théorie de la spécification, il apparaît alors naturel que certains travaux aient cherché à étudier cette propriété dans un cadre logique-indépendant (*cf.* [41, 42, 114, 46, 85, 92, 138]). Une telle étude a notamment permis de donner un énoncé plus général de l'interpolation de Craig en termes de morphismes de signatures et d'ensembles de formules. De plus, plusieurs formes d'interpolation, telles que l'interpolation de Craig-Robinson [1], ont ainsi pu être différenciées.

Dans la première section de ce chapitre, nous donnons un bref historique de l'interpolation de Craig depuis sa formulation originelle pour la théorie standard des modèles jusqu'à sa généralisation dans le cadre de la théorie abstraite des modèles. Nous expliquons pourquoi cette dernière formulation est plus adaptée à la théorie de la spécification. Nous donnons ensuite deux preuves institution-

1. L'interpolation de Craig-Robinson est une extension de l'interpolation de Craig considérant un ensemble secondaire de prémisses (*cf.* paragraphe « Interpolation de Craig-Robinson », section 2.1.1).

indépendantes de l'interpolation de Craig dont les domaines d'application sont complémentaires. La première, donnée par R. Diaconescu dans [41], s'applique aux logiques dont les classes de modèles sont fermées par certaines opérations algébriques (sous-structure, produit, *etc.*) mais ne possédant pas nécessairement la négation. La deuxième preuve, qui impose la négation, généralise la traditionnelle équivalence entre l'interpolation de Craig et le lemme de consistance de Robinson (établie dans le cadre de la logique du premier ordre classique). Nous nous inspirons d'une précédente preuve donnée par A. Salibra et G. Scollo dans [114] pour démontrer cette équivalence dans le cadre des institutions [2]. Enfin, dans la dernière section, nous donnons les conditions pour que l'interpolation de Craig soit préservée au travers d'un morphisme d'institutions. Cette preuve s'inspire également des travaux de A. Salibra et G. Scollo sus-mentionnés.

2.1.1 Historique

Le théorème d'interpolation est un résultat qui a été montré par W. Craig pour la logique du premier ordre classique (*cf.* [35]). C'est l'une des propriétés fondamentales de la théorie standard des modèles, équivalente au lemme de consistance de Robinson (*cf.* [30]) dans le cadre de la logique du premier ordre classique. Son énoncé, et donc sa preuve, est largement dépendant du cadre de cette logique.

Énoncé originel On considère un langage comme étant la donnée d'un ensemble de symboles logiques (connecteurs, quantificateurs, symboles de variables, symbole d'égalité) et, éventuellement, de symboles de fonctions et de prédicats. Les formules sont obtenues de façon habituelle à partir des formules atomiques construites sur les symboles de prédicats et le symbole d'égalité. Les modèles sont des structures du premier ordre et la relation de satisfaction des formules par les modèles est définie de manière usuelle (*cf.* exemple 1.1.3 et annexe B).

Théorème 2.1.1 (Interpolation de Craig ([35])) *Soient φ et ψ deux formules closes telles que $\varphi \models \psi$. Alors, il existe une formule close θ telle que :*

1. *$\varphi \models \theta$ et $\theta \models \psi$;*
2. *tout symbole de fonction ou de prédicat*[3] *apparaissant dans θ apparaît à la fois dans φ et dans θ.*

La formule θ est appelée interpolant *de φ et ψ.*

Ceci peut être reformulé de la façon suivante qui fait mieux apparaître la notion de changement de langage (*cf.* figure 2.1 ci-dessous) :

Soient $\mathcal{L}_1$ et $\mathcal{L}_2$ deux langages du premier ordre et soient $\varphi \in \mathcal{L}_1$ et $\psi \in \mathcal{L}_2$ deux formules closes telles que $\varphi \models \psi$ dans l'union de $\mathcal{L}_1$ et $\mathcal{L}_2$. Alors, il existe une formule θ appartenant à l'intersection de $\mathcal{L}_1$ et $\mathcal{L}_2$ telle que $\varphi \models \theta$ dans $\mathcal{L}_1$ et $\theta \models \psi$ dans $\mathcal{L}_2$.

2. Les travaux de A. Salibra et G. Scollo sur l'équivalence entre l'interpolation de Craig et le lemme de consistance de Robinson ont été réalisés dans le cadre restreint des pré-institutions.

3. L'égalité est un symbole distingué et n'est pas considérée comme un symbole de prédicat.

$$\begin{tikzcd} \mathcal{L}_1 \cap \mathcal{L}_2 \arrow[r, hook] \arrow[d, hook] & \mathcal{L}_1 \arrow[d, hook] \\ \mathcal{L}_2 \arrow[r, hook] & \mathcal{L}_1 \cup \mathcal{L}_2 \end{tikzcd}$$

FIG. 2.1 – *Inclusions de langages dans* FOL

Première généralisation Cette formulation (originelle) de l'interpolation de Craig est particulière aux mathématiques et à leur utilisation de la logique. Le cadre est celui de la logique du premier ordre avec égalité et les morphismes considérés sont restreints aux inclusions de langages. Or, pour les raisons évoquées en introduction, la notion de morphisme de signatures est d'une importance cruciale en théorie de la spécification car elle représente, de façon abstraite, la notion de modification d'interface d'un programme. Il est alors nécessaire de reformuler l'énoncé ci-dessus dans un cadre adapté à la théorie de la spécification, *i.e.* prenant acte de cette notion de morphisme. Le langage des institutions permet une telle formulation indépendamment du système logique sous-jacent.

Définition 2.1.1 (diagramme d'interpolation (formules)) *Soit une institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$. Un diagramme commutatif de morphismes de signatures*

$$\begin{tikzcd} \Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma_1'"] \\ \Sigma_2 \arrow[r, "\sigma_2'"'] & \Sigma' \end{tikzcd}$$

est un diagramme d'interpolation de Craig *si pour toute Σ_1-formule $\varphi \in \mathcal{S}en(\Sigma_1)$ et toute Σ_2-formule $\psi \in \mathcal{S}en(\Sigma_2)$ telles que $\mathcal{S}en(\sigma_1')(\varphi) \models_{\Sigma'} \mathcal{S}en(\sigma_2')(\psi)$, il existe une Σ-formule $\theta \in \mathcal{S}en(\Sigma)$ telle que $\varphi \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\theta)$ et $\mathcal{S}en(\sigma_2)(\theta) \models_{\Sigma_2} \psi$. La formule est appelée* interpolant *de φ et ψ.*
On dira que l'institution $\mathcal{I}$ possède la propriété d'interpolation de Craig pour les formules *pour une classe C de diagrammes commutatifs de morphismes de signatures si et seulement si tout diagramme de C est un diagramme d'interpolation de Craig.*

Le diagramme de morphismes de signatures ci-dessus généralise les traditionnelles opérations ensemblistes d'intersection et d'union. Selon les énoncés, les hypothèses considérées sont plus ou moins fortes (les diagrammes doivent être des pushouts [*cf.* [129, 21, 20, 47, 114]], des inclusions abstraites [*cf.* [45]], *etc.*) mais elles ne sont nécessaires que dans le sens où elles facilitent les preuves.

Deuxième généralisation Cette première généralisation de l'interpolation de Craig n'est toujours pas satisfaisante d'un point de vue théorie de la spécification. En effet, pour des problèmes d'expressivité, la théorie de la spécification fait usage de nombreuses logiques pour lesquelles cet énoncé n'est pas valable. Par exemple, il a été montré dans [13] que la logique équationnelle ne vérifie pas l'interpolation de Craig telle qu'énoncée dans la définition 2.1.1. En effet, ce résultat requiert nécessairement la compacité et la conjonction finie que ne possède pas la logique équationnelle. Or, d'un point de vue sémantique, il est équivalent de considérer un ensemble de formules ou une conjonction de formules. Nous adopterons donc l'énoncé suivant pour l'interpolation de Craig.

Définition 2.1.2 (diagramme d'interpolation (ensembles)) *Soit une institution $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ quelconque. Un diagramme commutatif de morphismes de signatures*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle \sigma_2}\downarrow & & \downarrow{\scriptstyle \sigma'_1} \\ \Sigma_2 & \xrightarrow[\sigma'_2]{} & \Sigma' \end{array}$$

est un diagramme d'interpolation de Craig *si pour tout ensemble $E_1 \subseteq \mathcal{Sen}(\Sigma_1)$ de Σ_1-formules et tout ensemble $E_2 \subseteq \mathcal{Sen}(\Sigma_2)$ de Σ_2-formules tels que $\mathcal{Sen}(\sigma'_1)(E_1) \models_{\Sigma'} \mathcal{Sen}(\sigma'_2)(E_2)$, il existe un ensemble $E \subseteq \mathcal{Sen}(\Sigma)$ de Σ-formules tel que $E_1 \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(E)$ et $\mathcal{Sen}(\sigma_2)(E) \models_{\Sigma_2} E_2$. L'ensemble E est appelé* interpolant *de E_1 et E_2.*
On dira que l'institution $\mathcal{I}$ possède la propriété d'interpolation de Craig pour les ensembles *pour une classe C de diagrammes commutatifs de morphismes de signatures si et seulement si tout diagramme de C est un diagramme d'interpolation de Craig.*

Notons que dans une institution compacte, si l'ensemble de formules E_2 est fini, l'ensemble interpolant E peut être choisi fini. La conséquence immédiate de ceci est que dans une institution compacte possédant la conjonction finie, cette dernière formulation de l'interpolation de Craig est équivalente à celle donnée à la définition 2.1.1 pour les formules.

Ce dernier énoncé institution-indépendant de l'interpolation de Craig a été proposé dans [41] par R. Diaconescu. En accord avec [110], [45] et [114], nous pensons que cela est plus naturel pour la théorie de la spécification. Cet énoncé généralise en effet les précédentes formulations institution-indépendantes de l'interpolation de Craig et ne limite pas l'étude de l'interpolation au cas d'inclusions de signatures. Notons qu'entre autres, les formulations proposées dans [114] (pour les formules) et dans [21] (versions forte et faible) sont des cas particuliers de cet énoncé. La seule hypothèse (inévitable) est que les morphismes de signatures considérés forment un diagramme commutatif. La raison permettant de se passer de la condition de pushout sur le diagramme de morphismes de signatures est exposée dans la remarque suivante.

Remarque 2.1.1 *Soit $\mathcal{I}$ une institution et soit le diagramme commutatif de la figure 2.2. Supposons que le quadruplet $(\Sigma, \Sigma_1, \Sigma_2, \Sigma'')$ soit un diagramme d'amalgation. Alors, le quadruplet $(\Sigma, \Sigma_1, \Sigma_2, \Sigma')$ est un diagramme d'amalgation faible si et seulement si σ est* conservatif, *i.e. tout Σ''-modèle admet une extension le long de σ.*

Dans les applications, sauf en de rares exceptions, les institutions considérées sont semi-exactes et les signatures admettent des pushouts. Alors, étant donné un diagramme commutatif de morphismes de signatures $(\Sigma, \Sigma_1, \Sigma_2, \Sigma')$, on considère le pushout $(\Sigma, \Sigma_1, \Sigma_2, \Sigma'')$ et l'unique morphisme de signatures

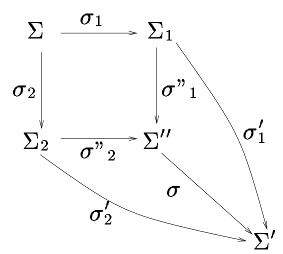

FIG. 2.2 –

$\sigma : \Sigma'' \to \Sigma'$. Par la propriété de semi-exactitude, le diagramme $(\Sigma, \Sigma_1, \Sigma_2, \Sigma'')$ est un diagramme d'amalgation, ce qui implique que $(\Sigma, \Sigma_1, \Sigma_2, \Sigma')$ est un diagramme d'amalgation faible si et seulement si σ est conservatif. Dans ce cas, pour tout couple de modèles $(\mathcal{M}_1, \mathcal{M}_2) \in |\mathcal{M}od(\Sigma_1)| \times |\mathcal{M}od(\Sigma_2)|$ tel que $\mathcal{M}od(\sigma_1)(\mathcal{M}_1) = \mathcal{M}od(\sigma_2)(\mathcal{M}_2)$, il existe alors un Σ'-modèle $\mathcal{M}'$ tel que $\mathcal{M}od(\sigma'_1)(\mathcal{M}') = \mathcal{M}'_1$ et $\mathcal{M}od(\sigma'_2)(\mathcal{M}') = \mathcal{M}_2$.

Interpolation de Craig-Robinson Les différents travaux menés dans le cadre des π-institutions et des logiques générales ont permis de mieux identifier les liens existant entre l'interpolation de Craig et la modularité. En particulier, ils ont mis en évidence l'existence d'autres formes d'interpolation, également liées à la modularité et pouvant être vérifiées lorsque l'interpolation classique ne l'est pas. Ceci est principalement vrai pour la forme alternative la plus répandue de l'interpolation de Craig : l'interpolation de Craig-Robinson.

Définition 2.1.3 (diagramme d'interpolation (Craig-Robinson)) *Soit une institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ quelconque. Un diagramme commutatif de morphismes de signatures*

$$\begin{tikzcd} \Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma'_1"] \\ \Sigma_2 \arrow[r, "\sigma'_2"'] & \Sigma' \end{tikzcd}$$

est un diagramme d'interpolation de Craig-Robinson *si pour tout ensemble $E_1 \subseteq \mathcal{S}en(\Sigma_1)$ de Σ_1-formules, tous ensembles $E_2, \Gamma_2 \subseteq \mathcal{S}en(\Sigma_2)$ de Σ_2-formules tels que $\mathcal{S}en(\sigma'_1)(E_1) \cup \mathcal{S}en(\sigma'_2)(\Gamma_2) \models_{\Sigma'} \mathcal{S}en(\sigma'_2)(E_2)$, il existe un ensemble $E \subseteq \mathcal{S}en(\Sigma)$ de Σ-formules tel que $E_1 \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(E)$ et $\Gamma_2 \cup \mathcal{S}en(\sigma_2)(E) \models_{\Sigma_2} E_2$. L'ensemble E est appelé* interpolant *de E_1 et E_2.*
On dira que l'institution $\mathcal{I}$ possède la propriété d'interpolation de Craig-Robinson *pour une classe C de diagrammes commutatifs de morphismes de signatures si et seulement si tout diagramme de C est un diagramme d'interpolation de Craig-Robinson.*

L'interpolation de Craig-Robinson renforce l'interpolation de Craig en distinguant un ensemble de prémisses « primaires » (l'ensemble E_1 ci-dessus) et un ensemble de prémisses « secondaires » (l'ensemble Γ_2 ci-dessus). On remarque aisément que tout diagramme de Craig-Robinson est un diagramme de Craig où l'ensemble Γ_2 est vide. La proposition suivante donne une condition suffisante pour que ces deux formes d'interpolation soient équivalentes.

Proposition 2.1.1 ([43]) *Pour toute institution possédant l'implication et, soit la compacité, soit la conjonction infinie, un diagramme commutatif de morphismes de signatures est un diagramme de Craig-Robinson si et seulement si c'est un diagramme de Craig.*

Un tel résultat est très intéressant car l'interpolation de Craig n'est pas suffisante en elle-même pour la modularisation. C'est une condition nécessaire mais pas suffisante.

2.1.2 Intérêt pour la théorie de la spécification

Nous montrons en détail dans cette section l'importance de l'interpolation de Craig pour la structuration et la modularité des spécifications axiomatiques présentées respectivement dans les sections 1.5 et 1.7.

Structuration

Nous avons vu en section 1.5 la généralisation que T. Borzyszkowski a donnée dans [21] de la structuration des spécifications de type axiomatique. Outre la définition d'un ensemble d'opérateurs de structuration des spécifications, ce travail identifie les conditions suffisantes pour que le système formel défini sur le système logique sous-jacent puisse être étendu aux spécifications structurées de façon correcte et complète. Il apparaît alors que l'interpolation de Craig est une des conditions requises pour obtenir cette complétude. Seulement, dans la pratique, l'interpolation de Craig n'est pas une propriété facile à démontrer et peut même n'être vérifiée, pour certaines logiques, que sur des diagrammes particuliers de morphismes de signatures. Ainsi, dans le cadre de la logique du premier ordre multi-sorte, T. Borzyszkowski a donné dans [20] les deux contre-exemples suivants (*cf.* exemples 1.1.4 et 1.1.5 pour une présentation des institutions PFOL et SUBPFOL).

Contre-exemple 2.1.1 *Soient les (*PFOL*-)signatures du premier ordre multi-sortes partielles suivantes :*
- $\Sigma = (\{s_1, s_2\}, \{a : s_1, b : s_2\}, \varnothing, \varnothing)$;
- $\Sigma_1 = (\{s\}, \{a : s\}, \varnothing, \varnothing)$;
- $\Sigma_2 = (\{s\}, \{a, b : s\}, \varnothing, \varnothing)$;
- $\Sigma' = \Sigma_1$.

Soit le pushout suivant dans $\mathcal{S}ig$

$$\begin{tikzcd} \Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma_1'"] \\ \Sigma_2 \arrow[r, "\sigma_2'"'] & \Sigma' \end{tikzcd}$$

tel que :
- $\sigma_1(s_1) = \sigma_1(s_2) = s$, $\sigma_1(a) = \sigma_1(b) = a$;
- $\sigma_2(s_1) = \sigma_2(s_2) = s$, $\sigma_2(a) = a$, $\sigma_2(b) = b$;

– $\sigma'_2(s) = s$, $\sigma'_2(a) = a$;
– $\sigma'_1(s) = s$, $\sigma'_1(a) = \sigma'_1(b) = a$,

Considérons alors le jugement suivant: $\mathcal{Sen}(\sigma'_2)(a \stackrel{e}{=} a) \models_{\Sigma'} \mathcal{Sen}(\sigma'_1)(a \stackrel{e}{=} b)$. *Ce jugement est satisfait par tout* Σ'*-modèle* $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$. *Supposons maintenant qu'il existe une* Σ*-formule* $\varphi \in \mathcal{Sen}(\Sigma)$ *telle que* $a \stackrel{e}{=} a \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(\varphi)$ *et* $\mathcal{Sen}(\sigma_2)(\varphi) \models_{\Sigma_2} a \stackrel{e}{=} b$. *Soit* $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ *un* Σ_2*-modèle ne satisfaisant pas la formule* $a \stackrel{e}{=} b$. *Cela signifie alors que l'interprétation de* a *dans* $\mathcal{M}_2$ *est différente de celle de* b, *i.e.* $a^{\mathcal{M}_2} \neq b^{\mathcal{M}_2}$. *Le réduit de* $\mathcal{M}_2$ *suivant* σ_2 *comporte deux ensembles de base, un pour chaque sorte. Ces deux ensembles sont tous deux égaux à l'ensemble de base de* $\mathcal{M}_2$. *Soit* $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$ *un* Σ*-modèle identique au réduit de* $\mathcal{M}_2$ *par* σ_2 *mais pour lequel* $a^{\mathcal{M}} = b^{\mathcal{Mod}(\sigma_2)(\mathcal{M}_2)}$. *Soit* $\mathcal{M}_1 \in |\mathcal{Mod}(\Sigma_1)|$ *l'unique* Σ_1*-modèle dont le réduit par* σ_1 *est égal à* $\mathcal{M}$.
Maintenant, puisque $\mathcal{M}_1$ *satisfait la formule* $a \stackrel{e}{=} a$, *il satisfait également la formule* $\mathcal{Sen}(\sigma_1)(\varphi)$. *Par la condition de satisfaction, on obtient que* $\mathcal{M}$ *satisfait* φ. *Ensuite, puisque* $\mathcal{M}$ *est isomorphe au réduit de* $\mathcal{M}_2$ *par* σ_2, *on a* $\mathcal{Mod}(\sigma_2)(\mathcal{M}_2) \models_{\Sigma} \varphi$. *Par la condition de satisfaction, on a alors* $\mathcal{M}_2 \models_{\Sigma_2} \mathcal{Sen}(\sigma_2)(\varphi)$, *ce qui contredit l'hypothèse que* $\mathcal{Sen}(\sigma_2)(\varphi) \models_{\Sigma_2} a \stackrel{e}{=} b$ *puisque* $a \stackrel{e}{=} b$ *n'est pas valide dans* $\mathcal{M}_2$.

On en déduit que si la logique du premier ordre avec égalité vérifie l'interpolation de Craig, ce n'est pas le cas de sa version multi-sorte. En effet, cette dernière peut être vue comme un fragment de la logique du premier ordre multi-sorte partielle. Or, dans le contre-exemple ci-dessus, il n'est pas fait usage de la partialité.
L'exemple suivant montre que la propriété d'interpolation n'est pas vérifiée par la logique du premier ordre multi-sorte partielle avec sous-sorte même si on ne considère que des morphismes injectifs à la fois sur les sortes et les noms de fonctions.

Contre-exemple 2.1.2 *Soit* $\Sigma = (\{s_1, s_2\}, \{a, b : s_1, c : s_2\}, \varnothing, \varnothing)$ *une (*SUBPFOL*-)signature du premier ordre multi-sorte partielle avec sous-sortes et soient* $\Sigma_1, \Sigma_2, \Sigma'$ *des* SUBPFOL*-signatures définies comme* Σ *sauf pour la relation de sous-sorte donnée par* $s_1 \leq_S s_2$. *On considère le même diagramme de morphismes de signatures que pour l'exemple précédent mais cette fois, les morphismes sont tous injectifs.*
Considérons alors le jugement suivant:

$$\mathcal{Sen}(\sigma'_2)(em_{s_1,s_2}(a) \stackrel{e}{=} c) \models_{\Sigma'} \mathcal{Sen}(\sigma'_1)(em_{s_1,s_2}(b) \stackrel{e}{=} c \Rightarrow a \stackrel{e}{=} b)$$

Ce jugement est satisfait par tout Σ'*-modèle. Supposons qu'il existe une* Σ*-formule* $\varphi \in \mathcal{Sen}(\Sigma)$ *telle que* $em_{s_1,s_2}(a) \stackrel{e}{=} c \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(\varphi)$ *et* $\mathcal{Sen}(\sigma_2)(\varphi) \models_{\Sigma_2} em_{s_1,s_2}(b) \stackrel{e}{=} c \Rightarrow a \stackrel{e}{=} b$. *Soit alors* $\mathcal{M}_1 \in |\mathcal{Mod}(\Sigma_1)|$ *un* Σ_1*-modèle tel que* $M_{1|s_1} = M_{1|s_2} = \{v_1, v_2\}$, *où* v_1 *et* v_2 *sont des valeurs différentes,* $em^{\mathcal{M}_1}_{s_1,s_2}$ *est l'identité,* $a^{\mathcal{M}_1} = c^{\mathcal{M}_1} = v_1$ *et* $b^{\mathcal{M}_1} = v_2$. *Soit* $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ *un* Σ_2*-modèle avec les mêmes ensembles de base et les mêmes interprétations des constantes* a, b *et* c *et tel que* $em^{\mathcal{M}_2}_{s_1,s_2}(v_1) = v_2$ *et* $em^{\mathcal{M}_2}_{s_1,s_2}(v_2) = v_1$. *Maintenant, puisque* $\mathcal{M}_1 \models_{\Sigma_1} em_{s_1,s_2}(a) \stackrel{e}{=} c$, *on a* $\mathcal{M}_1 \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(\varphi)$. *Par la condition de satisfaction, ceci est équivalent à* $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \models_{\Sigma} \varphi$. *Ensuite, puisque* $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}_2)$, *on a* $\mathcal{M}_2 \models_{\Sigma_2} \mathcal{Sen}(\sigma_2)(\varphi)$. *Finalement, par hypothèse* $\mathcal{M}_2 \models_{\Sigma_2} em_{s_1,s_2}(b) \stackrel{e}{=} c \Rightarrow a \stackrel{e}{=} b$, *ce qui est faux.*

Notons de plus que la propriété d'amalgation faible n'est pas vérifiée par l'institution de la logique du premier ordre multi-sorte partielle avec sous-sortes. En effet, il n'y a pas de modèle amalgamé pour les modèles $\mathcal{M}_1$ et $\mathcal{M}_2$ ci-dessus (*cf.* [60, 85, 84, 47] pour de plus amples informations sur les liens entre l'interpolation et l'amalgation (faible)). Sous certaines hypothèses pourtant, il est possible d'assurer la validité de l'interpolation de Craig comme le montre le théorème ci-dessous démontré par T. Borzyzskowski dans [20].

Définition 2.1.4 (extension par constantes) *Pour toute PFOL-signature Σ, l'*extension de Σ par des constantes *est la PFOL-signature $\Sigma^{\mathcal{C}}$ qui possède les mêmes ensembles de sortes, d'opérations partielles et totales et de prédicats et dont l'ensemble d'opérations totales est augmenté d'un ensemble infini dénombrable de symboles de constantes de sorte s pour chaque sorte s de Σ.*

On étend naturellement les morphismes de signatures aux signatures étendues par constantes. Pour tout morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ injectif sur les noms de sortes, on définit le morphisme $\sigma^{\mathcal{C}} : \Sigma_1^{\mathcal{C}} \to \Sigma_2^{\mathcal{C}}$ de la façon suivante :

Soient s_1 une sorte de Σ_1 et s_2 une sorte de Σ_2 telles que $\sigma(s_1) = s_2$. Soient $\mathcal{C}_1 = \{c_1^1 : s_1, c_2^1 : s_1, \dots\}$ et $\mathcal{C}_2 = \{c_1^2 : s_2, c_2^2 : s_2, \dots\}$ les ensembles dénombrables de constantes ajoutés respectivement à Σ_1 et Σ_2. On définit alors $\sigma^{\mathcal{C}}(c_i^1 : s_1) = c_i^2 : s_2$ pour tout $i \in \mathbb{N} \setminus \{0\}$.

On dit qu'une classe ζ de morphismes de signatures est *fermée par extension par constantes* si elle contient $\sigma^{\mathcal{C}}$ pour tout morphisme de signatures injectif σ et tout ensemble de contantes $\mathcal{C}$.

Théorème 2.1.2 (interpolation dans PFOL [20]) *L'institution* PFOL *possède la propriété d'interpolation lorsqu'on se restreint aux morphismes de signatures injectifs sur les noms de sortes et fermés par extension par constantes.*

Le résultat ci-dessus est également valable pour l'institution de la logique du premier ordre multi-sorte avec égalité.

Comme le montrent les deux exemples 2.1.1 et 2.1.2 ci-dessus, l'interpolation de Craig est une propriété qui est facilement mise en défaut. Pour certaines logiques, il se peut même que l'interpolation de Craig ne soit pas vérifiée de façon globale mais uniquement sur certains diagrammes particuliers. C'est ce que montre le théorème 2.1.2 en se restreignant aux morphismes de signatures injectifs sur les noms de sortes et fermés par extension par constantes. Dans [21], T. Borzyszkowski cherche à généraliser ce résultat et remarque que les morphismes de signatures servent au minimum à deux fins différentes :

1. Cacher des symboles dans la signature de la spécification cible ;
2. Ajouter ou renommer des symboles dans la spécification source.

Il introduit alors le cadre des $(\mathcal{D}, \mathcal{T})$-institutions pour y étudier les liens de l'interpolation de Craig avec la structuration et le raffinement. Intuitivement, une $(\mathcal{D}, \mathcal{T})$-institution est une institution possédant deux classes distinguées de morphismes de signatures contenant toutes deux l'identité et fermées par composition. De plus, pour tous morphismes $(d : \Sigma \to \Sigma_1) \in \mathcal{D}$ et $(t : \Sigma \to \Sigma_2) \in \mathcal{T}$, il existe $(t' : \Sigma_1 \to \Sigma') \in \mathcal{T}$ et $(d' : \Sigma_2 \to \Sigma') \in \mathcal{D}$ tels que le diagramme de la figure 2.3 soit un pushout dans $\mathcal{S}ig$. L'interpolation de Craig est alors

$$\begin{tikzcd}
& \Sigma \arrow[dl, "d"'] \arrow[dr, "t"] & \\
\Sigma_1 \arrow[dr, "t'"'] & & \Sigma_2 \arrow[dl, "d'"] \\
& \Sigma' &
\end{tikzcd}$$

FIG. 2.3 – $(\mathcal{D}, \mathcal{T})$-*pushout*

reformulée de la façon suivante dans le cadre des $(\mathcal{D}, \mathcal{T})$-institutions (*cf.* [21]):

> *une $(\mathcal{D}, \mathcal{T})$-institution vérifie la propriété de $(\mathcal{D}, \mathcal{T})$-interpolation si elle vérifie la propriété d'interpolation pour les formules (cf. définition 2.1.1) pour tout pushout de signatures tel que décrit sur la figure 2.3.*

On a alors les résultats suivants :

Théorème 2.1.3 (complétude [21]) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une $(\mathcal{D}, \mathcal{T})$-institution possédant la conjonction infinie et l'implication. Si :*

1. *$\mathcal{I}$ possède les propriétés de $(\mathcal{D}, \mathcal{T})$-interpolation et de $(\mathcal{D}, \mathcal{T})$-amalgation faible[4] ;*
2. *Les relations d'inférence $\vdash^{\mathcal{I}}_{\Sigma}$ sont complètes par rapport aux relations de conséquence sémantique $\models^{\mathcal{I}}_{\Sigma}$,*

alors, pour toute Σ-spécification (avec restriction $(\mathcal{D}, \mathcal{T})$) $SP \in \mathcal{S}pec(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$,

$$SP \models_\Sigma \varphi \Leftrightarrow SP \vdash_\Sigma \varphi$$

Dans le cas de $(\mathcal{D}, \mathcal{T})$-institutions compactes, le théorème précédent peut-être modifié comme suit.

Corollaire 2.1.1 ([21]) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une $(\mathcal{D}, \mathcal{T})$-institution compacte possédant la conjonction finie et l'implication. Si :*

1. *$\mathcal{I}$ possède les propriétés de $(\mathcal{D}, \mathcal{T})$-interpolation faible[5] et de $(\mathcal{D}, \mathcal{T})$-amalgation faible ;*
2. *Les relations d'inférence $\vdash^{\mathcal{I}}_{\Sigma}$ sont complètes par rapport aux relations de conséquence sémantique $\models^{\mathcal{I}}_{\Sigma}$;*
3. *$\mathcal{I}$ est compacte,*

alors, pour toute Σ-spécification $SP \in \mathcal{S}pec(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$,

$$SP \models_\Sigma \varphi \Leftrightarrow SP \vdash_\Sigma \varphi$$

4. La propriété de $(\mathcal{D}, \mathcal{T})$-amalgation faible est la propriété d'amalgation faible restreinte aux diagrammes tels qu'indiqués à la figure 2.3.

5. La propriéte de $(\mathcal{D}, \mathcal{T})$-interpolation faible est la propriété de $(\mathcal{D}, \mathcal{T})$-interpolation où on accepte un ensemble interpolant plutôt qu'une unique formule. Dans le cas d'institutions possédant la conjonction infinie, ces deux propriétés sont équivalentes.

Définition 2.1.5 (spécification finie) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et soit $\Sigma \in |\mathcal{S}ig|$ une signature. Une Σ-spécification (Σ, Φ) est dite finie si l'ensemble $\Phi \subseteq \mathcal{S}en(\Sigma)$ est fini, i.e. $\Phi \in \wp_f(\mathcal{S}en(\Sigma))$.*

Dans le cas de spécifications finies, on peut se passer de la propriété de compacité.

Corollaire 2.1.2 ([21]) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une $(\mathcal{D}, \mathcal{T})$-institution possédant la conjonction finie et l'implication. Si :*

1. *$\mathcal{I}$ possède les propriétés de $(\mathcal{D}, \mathcal{T})$-interpolation et de $(\mathcal{D}, \mathcal{T})$-amalgation faible ;*
2. *Les relations d'inférence $\vdash^{\mathcal{I}}_{\Sigma}$ sont complètes par rapport aux relations de conséquence sémantique $\models^{\mathcal{I}}_{\Sigma}$;*

alors, pour toute Σ-spécification finie $SP \in \mathcal{S}pec(\Sigma)$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$,

$$SP \models_{\Sigma} \varphi \Leftrightarrow SP \vdash_{\Sigma} \varphi$$

Le lemme suivant montre une conséquence de la complétude de la relation d'inférence $\vdash_{\Sigma} \subseteq \mathcal{S}pec(\Sigma) \times \mathcal{S}en(\Sigma)$ $(\Sigma \in |\mathcal{S}ig|)$ pour toute institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$.

Définition 2.1.6 (spécifications équivalentes) *Soient Σ une signature et $SP_1, SP_2 \in \mathcal{S}pec(\Sigma)$ deux spécifications. On dit que SP_1 et SP_2 sont* équivalentes, *noté $SP_1 \cong SP_2$ si :*

$$\mathcal{M}od(SP_1) = \mathcal{M}od(SP_2)$$

Lemme 2.1.1 ([21]) *Pour toute institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$, toute signature $\Sigma \in |\mathcal{S}ig|$, toutes Σ-spécifications $SP_1, SP_2 \in \mathcal{S}pec(\Sigma)$, toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, si la relation d'inférence $\vdash_{\Sigma} \subseteq \mathcal{S}pec(\Sigma) \times \mathcal{S}en(\Sigma)$ est complète par rapport à la relation $\models_{\Sigma} \subseteq \mathcal{S}pec(\Sigma) \times \mathcal{S}en(\Sigma)$ alors,*

$$SP_1 \cong SP_2 \Rightarrow (SP_1 \vdash_{\Sigma} \varphi \Leftrightarrow SP_2 \vdash_{\Sigma} \varphi)$$

Le théorème suivant montre l'importance de l'interpolation de Craig pour la complétude des systèmes de preuve compositionnels.

Théorème 2.1.4 ([21]) *Pour toute $(\mathcal{D}, \mathcal{T})$-institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ satisfaisant la propriété de $(\mathcal{D}, \mathcal{T})$-amalgation faible mais pas la propriété de $(\mathcal{D}, \mathcal{T})$-interpolation faible, le système logique présenté à la définition 1.6.6 n'est pas complet.*

Modularité

L'interpolation de Craig a récemment été étudiée sous sa forme généralisée d'interpolation de Craig-Robinson dans le cadre des systèmes d'inférence (*cf.* [47, 139, 138, 140]). Dans [47], les auteurs établissent l'équivalence entre cette dernière propriété et la stabilité des extensions conservatives par pushout (ce qui généralise le théorème de modularisation de [136, 139]). Ils montrent notamment que démontrer la stabilité par pushout d'un morphisme particulier de théories est équivalent à démontrer que la propriété de Craig-Robinson est vérifiée sur des diagrammes particuliers. Ceci ne suppose ou n'implique aucunement que l'interpolation de Craig-Robinson (ou toute autre forme d'interpolation) soit vérifiée globalement par le système logique considéré. Dans [140], puis dans [138] les auteurs poussent encore plus loin l'étude des relations entre l'interpolation de Craig-Robinson et la propriété de modularisation en considérant différentes formes de modularisation (présentation-modulaire, langage-modulaire, axiome-modulaire). Nous reprenons ici les résultats présentés dans [47].

Théorème 2.1.5 ([47]) *Pour tout diagramme décrivant un pushout de morphismes de théories l'interpolation de Craig-Robinson est une condition nécessaire et suffisante à la propriété de modularisation.*

On remarquera que le théorème précédent ne suppose pas que l'interpolation de Craig-Robinson soit vérifiée de façon globale. De plus, comme indiqué dans l'exemple suivant, il existe des systèmes d'inférence intéressants et non-triviaux, vérifiant l'interpolation de Craig-Robinson et la modularisation de façon globale mais pas l'interpolation de Craig.

Exemple 2.1.1 ([47]) *Le calcul de l'arithmétique du premier ordre possède l'interpolation de Craig-Robinson, et donc la modularisation, mais ne vérifie pas l'interpolation de Craig. Le théorème classique établi pour la logique du premier ordre ne s'applique pas ici car il peut exister des formules φ et ψ telles que $\varphi \Rightarrow \psi$ est obtenue par induction. Dans ce cas, l'axiome d'induction utilisé dépend de l'union des symboles de prédicats apparaissant dans φ et ψ. En effet, la preuve déductive standard du théorème d'interpolation de Craig échoue (dans un système de preuve à la Gentzen) à la règle $\frac{\ldots \alpha \Rightarrow \beta(n) \ldots}{\alpha \Rightarrow \forall x \beta(x)}$. Un interpolant θ_n étant identifié pour chaque prémisse $\alpha \Rightarrow \beta(n)$, il faut maintenant prendre la conjonction infinie de ces interpolants. Or, ceci n'est pas autorisé par la syntaxe (cf. [92] pour une preuve plus détaillée).*

2.2 Preuves institution-indépendantes

Comme nous venons de le voir dans la section précédente, l'interpolation de Craig est une propriété cruciale pour la théorie de la spécification. Il est donc d'un grand intérêt d'identifier les systèmes logiques possédant cette propriété afin de pouvoir profiter des résultats énoncés dans la section précédente. À notre connaissance, seuls R. Diaconescu dans [41] d'un côté et A. Salibra et G. Scollo dans [114] de l'autre ont étudié l'interpolation de Craig dans le paradigme institution-indépendant. Ces deux approches sont très différentes dans leur principe et leurs domaines d'application sont disjoints. En effet, le travail de A. Salibra et G. Scollo porte sur les institutions possédant nécessairement la négation. Dans le cadre de la théorie standard des modèles (donc en supposant la négation), l'interpolation de Craig est équivalente au lemme de consistance de Robinson qui assure que la consistance est préservée au travers de l'union de théories sous réserve que les théories considérées soient des extensions consistantes d'une même théorie complète. Dans [114], ils ont généralisé cette équivalence dans le cadre de la théorie abstraite des modèles et étudié les conditions de transfert de l'interpolation de Craig d'un système à un autre. De son côté, R. Diaconescu s'est intéressé aux logiques ne possédant pas nécessairement la négation et donc pour lesquelles il n'est pas possible d'utiliser le lemme de consistance de Robinson. Cependant, ces logiques possèdent généralement des classes de modèles stables par certaines opérations algébriques (quotient, produit, sous-structure, *etc.*). Dans [41], R. Diaconescu étend le cadre des institutions pour prendre en compte cette notion de fermeture des classes de modèles. Il introduit ainsi le cadre des institutions de Birkhoff et démontre la validité de l'interpolation de Craig pour tout système logique présentable par une institution de Birkhoff, moyennant quelques conditions très simples.

Nous commençons par présenter les institutions de Birkhoff et nous énonçons le résultat démontré par R. Diaconescu dans [41]. Dans la deuxième section, nous nous inspirons d'un résultat similaire de A. Salibra et G. Scollo (*cf.* [114]) pour démontrer l'équivalence de l'interpolation de Craig et du lemme de consistance de Robinson pour les institutions possédant la négation. Enfin, dans la dernière section, nous rappelons le résultat de A. Salibra et G. Scollo (*cf.* [114]) sur le transfert de l'interpolation de Craig via la notion de transformation d'institutions et nous proposons le même type de résultat mais pour les morphismes d'institutions.

2.2.1 Interpolation par axiomatisabilité

Afin d'introduire les institutions de Birkhoff, il nous faut rappeler certaines notions usuelles sur les filtres et ultrafiltres.

Définition 2.2.1 (filtre) *Soit E un ensemble non vide. Un* filtre *sur E est la donnée d'un ensemble $F \subseteq \wp(E)$ tel que :*

1. $E \in F$;
2. $(X \in F \wedge Y \in F) \Rightarrow X \cap Y \in F$;
3. $(X \in F \wedge X \subseteq Y) \Rightarrow Y \in F$.

On parlera de *filtre propre* lorsque F est strictement inclus dans $\wp(E)$ et d'*ultrafiltre* si F est un filtre vérifiant la condition suivante :

$$\forall X \in \wp(E),\ X \in F \Leftrightarrow (E \setminus X) \notin F$$

Les ultrafiltres sont bien entendu des cas particuliers de filtres propres.

La notion de filtre permet de donner une caractérisation catégorielle de la construction de produits réduits[6] s'appuyant sur la notion de colimites directes. On rappelle que :

- une *catégorie indexée* $\mathcal{C}$ est la donnée d'un foncteur $\mathcal{C} : I^{op} \to \mathcal{C}at$, où $I \in |\mathcal{S}et|$ est un ensemble arbitraire d'indices ;
- un ensemble partiellement ordonné $(J, \leq)$ est *dirigét* si pour tout couple d'éléments $(i, j) \in J^2$, il existe un élément $k \in J$ tel que $i \leq k$ et $j \leq k$;
- une colimite[7] d'un foncteur $D : J \to C$ ($C \in \mathcal{C}at$) est *dirigé* si J est un ensemble ordonné direct.

6. La construction de produits réduits, habituelle en théorie des modèles classique (*cf.* chapitre 4 de [30]) a probablement été formulée en termes catégoriels pour la première fois dans [89] et a depuis été utilisée dans quelques travaux en théorie des modèles tels que [6]. L'équivalence entre les versions catégorielle et ensembliste a été montrée dans [65].

7. *cf.* annexe A pour la définition de colimite.

Définition 2.2.2 (produit réduit) *Soit $\mathcal{C} : I^{op} \to \mathcal{C}at$ une catégorie indexée munie de produits et de colimites directes. Soit $\{A_i\}_{i\in I}$ une famile d'objets de $\mathcal{C}$. Tout filtre F sur l'ensembe d'indices I définit un foncteur $A_F : F \to C$ tel que : $\forall J, J' \in F, J \subseteq J'$,*

$$A_F(J \subseteq J') = p_{J',J} : \prod_{i\in J'} A_i \to \prod_{i\in J} A_i$$

où $p_{J',J}$ est la projection canonique.
Le produit réduit de la famille $\{A_i\}_{i\in I}$ modulo F *est alors la colimite $\mu : A_F \Rightarrow \prod_F A_i$ du foncteur A_F.*

$$\begin{tikzcd}
\prod_{i\in J'} A_i \arrow[rr, "p_{J',J}"] \arrow[dr, "\mu_{J'}"'] & & \prod_{i\in J} A_i \arrow[dl, "\mu_J"] \\
& \prod_F A_i &
\end{tikzcd}$$

Dans le cas où le filtre F est un ultrafiltre, le produit réduit modulo F est appelé un *ultraproduit.*

Notons que puisque F est un ensemble ordonné direct, le produit réduit modulo F existe toujours, sous réserve des hypothèses de la définition précédente.

Notation 2.2.1 *Soient $\mathcal{F}$ une classe de filtres et $\mathcal{C} : I^{op} \to \mathcal{C}at$ une catégorie indexée. Pour toute classe $K \subseteq |C|$ d'objets de C, notons $\mathcal{F}(K)$ la classe de tous les produits réduits modulo F ($F \in \mathcal{F}$) d'objets pris dans K, i.e. :*

$$\mathcal{F}(K) = \{\prod_F A_i / F \in \mathcal{F} \wedge (\forall i \in I,\, A_i \in K)\}$$

où I est l'ensemble d'indices sur lequel les filtres $F \in \mathcal{F}$ sont construits.

L'idée principale derrière les institutions de Birkhoff (*cf.* [41]) est que la fermeture axiomatique d'une classe de modèles $\mathscr{M}$ peut-être obtenue à partir de $\mathscr{M}$ suivant les deux étapes suivantes :

1. Prendre tous les produits réduits de $\mathscr{M}$ pour une classe de filtres particulière ;
2. Considérer tous les modèles partageant une certaine « relation de Birkhoff » avec au moins l'un de ces produits réduits.

Notation 2.2.2 *Afin de clarifier l'énoncé de la définition des institutions de Birkhoff, nous introduisons ponctuellement les notations suivantes, lesquelles sont redondantes avec celles données à la définition 1.3.1.*
Soient $(\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et $\Sigma \in |\mathcal{S}ig|$ une signature.

- *pour tout ensemble de formules $E \subseteq \mathcal{S}en(\Sigma)$, on notera E^* l'ensemble précédemment noté $\mathcal{M}od(E)$ des Σ-modèles satisfaisant E, i.e. :*

$$E^* = \{\mathcal{M} \in |\mathcal{M}od(\Sigma)| / \mathcal{M} \models_\Sigma E\}$$

- *pour toute classe* $\mathscr{M} \subseteq |\mathcal{Mod}(\Sigma)|$ *de modèles, on notera* $\mathscr{M}^*$ *l'ensemble précédemment noté* $\mathcal{Th}(\mathscr{M})$ *et appelé théorie de* $\mathscr{M}$*, i.e. :*

$$\mathscr{M}^* = \{\varphi \in \mathcal{Sen}(\Sigma)/\forall \mathcal{M} \in \mathscr{M},\ \mathcal{M} \models_\Sigma \varphi\}$$

On introduit également les notations suivantes relatives aux relations binaires :

- *si* $\mathcal{R} \subseteq A \times B$ $(A, B \in |\mathcal{Set}|)$ *est une relation binaire alors la relation inverse, notée* $\mathcal{R}^{-1}$*, est définie de la façon suivante :*

$$\mathcal{R}^{-1} = \{(b, a)/(a, b) \in R\}$$

- *si* $\mathcal{R} \subseteq A \times B$ $(A, B \in |\mathcal{Set}|)$ *est une relation binaire alors, pour tout* $A' \subseteq A$*, on note* $R(A')$ *l'ensemble des classes d'équivalence des éléments de* A'*, i.e. :*

$$\mathcal{R}(A') = \{b/(a, b) \in \mathcal{R}, a \in A'\}$$

Définition 2.2.3 (institution de Birkhoff) *Le sextuplet* $(\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models, \mathcal{F}, \mathcal{B})$ *est une* institution de Birkhoff *si et seulement si :*

- $(\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ *est une* institution *telle que pour toute signature* $\Sigma \in |\mathcal{Sig}|$*,* $\mathcal{Mod}(\Sigma)$ *est une catégorie indexée munie de produits et de colimites directes ;*
- $\mathcal{F}$ *est une* classe de filtres *avec* $\{\{\{*\}\}\} \in \mathcal{F}$ *;*
- $\mathcal{B}_\Sigma \subseteq |\mathcal{Mod}(\Sigma)| \times |\mathcal{Mod}(\Sigma)|$ *est une* relation binaire et réflexive *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *telle que :*

$$\forall \mathscr{M} \subseteq |\mathcal{Mod}(\Sigma)|,\ \mathscr{M}^{**} = \mathcal{B}_\Sigma^{-1}(\mathcal{F}(\mathscr{M}))$$

Remarquons les deux faits suivants. Si $\mathcal{F}$ une classe de filtres et C une catégorie alors, pour toute classe $K \subseteq |C|$ d'objets de C $\mathcal{F}(K)$ est :

1. La fermeture de K par produits lorsque $\mathcal{F} = \{\{I\}/I \in |\mathcal{Set}|\}$;
2. La fermeture de K par isomorphismes lorsque $\mathcal{F} = \{\{\{*\}\}\}$.

Exemple 2.2.1 *Soit* $\mathcal{C}$ *une catégorie munie d'un système d'inclusion (cf. définition 3.1.1). On a alors les résultats suivants :*

- $a \in |\mathcal{C}|$ *est un sous-objet de* $b \in |\mathcal{C}|$ *s'il existe une inclusion abstraite* $a \hookrightarrow b$ *dans* $\mathcal{C}$ *;*
- *un objet* $b \in |\mathcal{C}|$ *est une représentation quotiente d'un objet* $a \in |\mathcal{C}|$ *s'il existe une surjection abstraite* $a \to b$ *dans* $\mathcal{C}$*. Un quotient de* a *est une classe isomorphe de représentation quotiente.*

D'après le classique théorème de Birkhoff sur les variétés, pour toute signature algébrique Σ*, une classe de* Σ*-algèbres* $\mathcal{A}$ *est une variété si et seulement si il existe un ensemble de* Σ*-équations universelles tel que* $\mathcal{A} = E^*$*. L'exemple le plus classique d'institution de Birkhoff est alors une algèbre générale dont les formules sont des équations. Par le théorème des variétés,* $\mathcal{B} = \xrightarrow{S} \circ \xleftarrow{H}$*, où* H *est la classe des homomorphismes d'algèbres surjectifs et* S *est la classe des homomorphismes d'algèbres injectifs, et* $\mathcal{F} = \{\{E\}/E \in |\mathcal{Set}|\}$*.*

Exemple 2.2.2 *Par le théorème des quasi-variétés, pour l'institution des algèbres munies d'équations conditionnelles infinies*[8]*,* $\mathcal{B} = \xrightarrow{S}$ *et* $\mathcal{F}$ *comme précédemment, alors que pour l'institution des algèbres munies d'équations conditionnelles finies*[9]*,* $\mathcal{B} = \xrightarrow{S}$ *et* $\mathcal{F}$ *est la classe de tous les filtres.*

8. La condition de l'équation est une conjonction infinie d'équations.
9. La condition de l'équation est une conjonction finie d'équations.

Notons que même la logique du 1^{er} ordre (mono- et multi-sorte) peut être présentée par une institution de Birkhoff en prenant $\mathcal{F}$ comme la classe de tous les ultrafiltres et $\mathcal{B}$ comme la relation d'équivalence élémentaire (*cf.* [30]). De façon similaire, mais plus générale, toute institution de Lós (*cf.* [39]) possédant la négation et la conjonction (finie) peut être représentée par une institution de Birkhoff (*cf.* [43] pour de plus nombreux exemples d'institutions de Birkhoff).

On rappelle qu'une *catégorie indexée* est la donnée d'un foncteur $C : I^{op} \to \mathcal{C}at$, où I est un ensemble arbitraire d'indices.

Définition 2.2.4 *Soit $C : I^{op} \to \mathcal{C}at$ une catégorie indexée et soit $R = \{R_i \subseteq |C_i|^2 / i \in |I|\}$ une relation binaire indexée par I. On dit qu'une flèche $u \in I(i,i')$* élève *R si et seulement si pour tout objet $M' \in |C_{i'}|$ et tout objet $N \in |C_i|$, si $(C_u(M'), N) \in R_i$ alors il existe un objet $N' \in |C_{i'}|$ tel que $C_u(N') = N$ et $(M', N') \in R_{i'}$.*

Théorème 2.2.1 ([41]) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models, \mathcal{F}, \mathcal{B})$ une institution de Birkhoff et soit un diagramme d'amalgation faible dans $\mathcal{I}$ comme suit*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle \sigma_2}\downarrow & & \downarrow{\scriptstyle \sigma'_1} \\ \Sigma_2 & \xrightarrow[\sigma'_2]{} & \Sigma' \end{array}$$

et tel que :

1. *$\mathcal{M}od(\sigma_1)$ préserve les petits produits et les colimites directes ;*
2. *σ_2 élève $\mathcal{B}$.*

Alors, ce diagramme est un diagramme d'interpolation de Craig.

2.2.2 Interpolation par consistance

Le lemme de consistance de Robinson (*cf.* [109]) est un des résultats fondamentaux de la logique moderne pour les langages du premier ordre. Il a été montré indépendemment par W. Craig (*cf.* [35]) sous la forme équivalente du théorème d'interpolation dans le cadre de la logique du premier ordre classique (en supposant donc la négation). Nous généralisons ici cette équivalence entre l'interpolation de Craig et le lemme de consistance de Robinson (ou plus exactement la propriété de Robinson). La preuve que nous donnons est inspirée de celle donnée par A. Salibra et G. Scollo dans un cadre (pré-)institution-indépendant [10] (*cf.* [114]). Elle en constitue une amélioration dans le sens où nous ne requerons que la négation [11].

10. Intuitivement, les pré-institutions (*cf.* [112]) sont un affaiblissement du cadre des institutions dans le sens où elles n'imposent ni que les classes de modèles forment des catégories ni que la condition de satisfaction soit vérifiée.

11. Dans [114] les auteurs imposaient également la propriété d'expansion élémentaire, *i.e.* pour tout morphisme de signatures $\sigma \in \mathcal{S}ig(\Sigma_1, \Sigma_2)$, toute Σ_2-présentation $E \subseteq \mathcal{S}en(\Sigma_2)$ et tout Σ_1-modèle $\mathcal{M}_1 \in |\mathcal{M}od(\Sigma_1)|$, on a :

$$\mathcal{M}_1 \models_{\Sigma_1} E_\sigma \Rightarrow (\exists \mathcal{M}_2 \in |\mathcal{M}od(\Sigma_2)|, \mathcal{M}_2 \models_{\Sigma_2} E \wedge \mathcal{M}_1 \equiv_\Sigma \mathcal{M}od(\sigma)(\mathcal{M}_2))$$

où $E_\sigma = \{\varphi \in \mathcal{S}en(\Sigma_1) / E \models_{\Sigma_2} \mathcal{S}en(\sigma)(\varphi)\}$.

Lemme 2.2.1 *Soit le diagramme commutatif de morphismes de signatures suivant :*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle\sigma_2}\downarrow & & \downarrow{\scriptstyle\sigma_1'} \\ \Sigma_2 & \xrightarrow[\sigma_2']{} & \Sigma' \end{array}$$

Soit $E_1 \subseteq \mathcal{S}en(\Sigma_1)$ une Σ_1-théorie consistante et soit $\mathcal{M} \in |\mathcal{M}od(\Sigma_2)|$ un Σ_2-modèle tel que $\mathcal{M} \models_{\Sigma_2} \mathcal{S}en(\sigma_2)(\mathcal{T}h_\Sigma(E_1))$, où $\mathcal{T}h_\Sigma(E_1) = \{\varphi \in \mathcal{S}en(\Sigma / E_1 \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\varphi))\}$. Alors, $E_1 \cup \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$ est une Σ_1-théorie consistante.

Preuve Par la condition de satisfaction, puisque $\mathcal{M} \models_{\Sigma 2} \mathcal{S}en(\sigma_2)(\mathcal{T}h_\Sigma(E_1))$, on a $\mathcal{M}od(\sigma_2)(\mathcal{M}) \models_\Sigma \mathcal{T}h_\Sigma(E_1)$. On en déduit que $\mathcal{T}h_\Sigma(E_1) \cup \mathcal{T}h_\Sigma(\mathcal{M})$ est une Σ-théorie consistante puisque $\mathcal{M}od(\sigma_2)(\mathcal{M})$ en est un modèle. Or, $\mathcal{T}h_\Sigma(\mathcal{M})$ est une Σ-théorie complète. On en déduit que $\mathcal{T}h_\Sigma(E_1) \subseteq \mathcal{T}h_\Sigma(\mathcal{M})$. Supposons maintenant que $E_1 \cup \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$ soit inconsistante et soit $\varphi \in \mathcal{S}en(\Sigma)$. Alors, $E_1 \cup \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M})) \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\varphi)$ et $E_1 \cup \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M})) \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\neg\varphi)$. Puique E_1 et $\mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$ sont des Σ_2-théories consistantes, il y a deux cas à traiter :

1. $\mathcal{S}en(\sigma_1)(\varphi) \in E_1$ et $\neg\mathcal{S}en(\sigma_1)(\varphi) \in \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$:
   Par définition, ceci signifie que $\varphi \in \mathcal{T}h_\Sigma(E_1)$ et donc $\varphi \in \mathcal{T}h_\Sigma(\mathcal{M})$. Puisque cette dernière théorie est consistante, on en déduit que $\neg\varphi \notin \mathcal{T}h_\Sigma(\mathcal{M})$ et donc que $\mathcal{S}en(\sigma_1)(\neg\varphi) \notin \mathcal{S}en(\sigma_1)(\mathcal{T}h(\mathcal{M}))$
2. $\mathcal{S}en(\sigma_1)(\varphi) \in \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$ et $\neg\mathcal{S}en(\sigma_1)(\varphi) \in E_1$:
   Par définition, ceci signifie que $\varphi \in \mathcal{T}h_\Sigma(\mathcal{M})$ et donc $\neg\varphi \notin \mathcal{T}h_\Sigma(\mathcal{M})$. On en déduit que $\neg\varphi \notin \mathcal{T}h_\Sigma(E_1)$ et donc que $\neg\varphi \notin E_1$.

□

Théorème 2.2.2 *Pour toute institution $\mathcal{I}$ fermée par la négation, l'interpolation de Craig sur les formules est équivalente à la propriété de Robinson.*

Preuve Soit le diagramme commutatif de morphismes de signatures suivant :

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle\sigma_2}\downarrow & & \downarrow{\scriptstyle\sigma_1'} \\ \Sigma_2 & \xrightarrow[\sigma_2']{} & \Sigma' \end{array}$$

PR⇒PIC : Soient $E_i \subseteq \mathcal{S}en(\Sigma_i)$ $(i \in \{1,2\})$ des ensembles de formules tels que :

$$\mathcal{S}en(\sigma_1')(E_1) \models_{\Sigma'} \mathcal{S}en(\sigma_2')(E_2)$$

Soit $\mathcal{T}h_\Sigma(E_1) = \{\varphi \in \mathcal{S}en(\Sigma / E_1 \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(E_1))\}$. Supposons que $\mathcal{S}en(\sigma_2)(\mathcal{T}h_\Sigma(E_1)) \not\models_{\Sigma_2} E_2$. Alors, par définition, il existe un Σ_2-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma_2)|$ et une Σ_2-formule $\psi \in E_2$ tels que $\mathcal{M} \in \mathcal{M}od(\mathcal{S}en(\sigma_2)(\mathcal{T}h_\Sigma(E_1)) \cup \{\neg\psi\})$. Soit $\mathcal{T}h_\Sigma(\mathcal{M}) = \{\varphi \in \mathcal{S}en(\Sigma) / \mathcal{M}od(\sigma_2)(\mathcal{M}) \models_\Sigma \varphi\}$. Cette Σ-théorie est évidemment complète et, de plus, $\mathcal{S}en(\sigma_2)(Th_\Sigma(\mathcal{M})) \cup \{\neg\psi\}$ est une Σ_2-théorie consistante. Par le lemme 2.2.1, $E_1 \cup \mathcal{S}en(\sigma_1)(\mathcal{T}h_\Sigma(\mathcal{M}))$ est également une Σ_1-théorie consistante. Par le lemme de consistance de Robinson, on a alors $\mathcal{S}en(\sigma_1' \circ$

$\sigma_1)(Th_\Sigma(\mathcal{M})) \cup \mathcal{S}en(\sigma'_1)(E_1) \cup \{\neg \mathcal{S}en(\sigma'_2)(\psi)\}$ est consistant. Mais ceci contredit le fait que $\mathcal{S}en(\sigma'_1)(E_1) \models_{\Sigma'} \mathcal{S}en(\sigma'_2)(E_2)$.

PIC⇒PR : Soient $\mathcal{M}_i \in |\mathcal{M}od(\Sigma_i)|$ $(i \in \{1,2\})$ tels que $\mathcal{M}od(\sigma_1)(\mathcal{M}_1) \equiv_\Sigma \mathcal{M}od(\sigma_2)(\mathcal{M}_2)$. Puisque $\mathcal{I}$ est fermée par la négation, $\mathcal{T}h(\mathcal{M}od(\sigma_i)(\mathcal{M}_i))$ et $\mathcal{T}h(\mathcal{M}_i)$ $(i \in \{1,2\})$ sont des théories complètes. Si $\bigcup_{i=1,2} \mathcal{S}en(\sigma'_i)(\mathcal{T}h(\mathcal{M}_i))$ est consistant alors, pour tout modèle $\mathcal{M}' \in \mathcal{M}od(\bigcup_{i=1,2} \mathcal{S}en(\sigma'_i)(\mathcal{T}h(\mathcal{M}_i)))$, nous avons par la condition de satisfaction que $\mathcal{M}od(\sigma'_i)(\mathcal{M}') \in \mathcal{M}od(\mathcal{T}h(\mathcal{M}_i))$. Puisque $Th(\mathcal{M}_i)$ est une théorie complète, on a $\mathcal{M}od(\sigma'_i)(\mathcal{M}') \equiv_{\Sigma_i} \mathcal{M}_i$.
Il ne reste plus qu'à montrer que $\bigcup_{i=1,2} \mathcal{S}en(\sigma'_i)(\mathcal{T}h(\mathcal{M}_i))$ est consistante. Supposons que ce ne soit pas le cas. Alors, il existe une formule $\varphi \in \mathcal{T}h(\mathcal{M}_2)$ telle que $\mathcal{S}en(\sigma'_1)(\mathcal{T}h(\mathcal{M}_1)) \models_{\Sigma'} \mathcal{S}en(\sigma'_2)(\neg\varphi)$. Par le théorème d'interpolation de Craig, il existe un ensemble de formules $E \subseteq \mathcal{S}en(\Sigma)$ tel que $\mathcal{T}h(\mathcal{M}_1) \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(E)$ et $\mathcal{S}en(\sigma_2)(E) \models_{\Sigma_2} \neg\varphi$. Puisque $\mathcal{T}h(\mathcal{M}_1)$ est consistante, pour toute formule $\psi \in E$ on a $\mathcal{T}h(\mathcal{M}_1) \models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\psi)$ et donc $\mathcal{T}h(\mathcal{M}_1) \not\models_{\Sigma_1} \mathcal{S}en(\sigma_1)(\neg\psi)$. De plus, il existe une formule $\psi \in E$ telle que $\mathcal{T}h(\mathcal{M}_2) \models_{\Sigma_2} \mathcal{S}en(\sigma_2)(\neg\psi)$ et donc $\mathcal{T}h(\mathcal{M}_2) \not\models_{\Sigma_2} \mathcal{S}en(\sigma_2)(\psi)$. Comme les rôles de $\mathcal{M}_1$ et $\mathcal{M}_2$ ci-dessus sont interchangeables, on en déduit qu'il existe une formule $\psi \in E$ telle que $\mathcal{T}h(\mathcal{M}od(\sigma_i)(\mathcal{M}_i) \not\models \neg\psi$ et $\mathcal{T}h(\mathcal{M}od(\sigma_i)(\mathcal{M}_i) \not\models \psi$. Ceci contredit le fait que $\mathcal{T}h(\mathcal{M}od(\sigma_i)(\mathcal{M}_i)) \models \psi$ est une théorie complète.

□

2.3 Préservation de l'interpolation de Craig

2.3.1 Préservation par transformation d'institutions

Lorsqu'on introduit un nouveau formalisme, il est agréable de pouvoir profiter directement d'outils et de techniques d'aide au développement sans avoir à les redévelopper. Cependant, les différences avec les formalismes existants peuvent être suffisamment importantes pour empêcher le transfert direct de ces outils et techniques. Or, la définition de ces derniers repose sur un certain nombre de résultats généraux tels que l'interpolation de Craig par exemple. Plutôt que de devoir redémontrer ces résultats, il peut être souhaitable d'établir des relations avec des systèmes logiques existants et possédant les propriétés souhaitées. Sous certaines hypothèses quant à la nature de ces relations, il sera alors possible de transférer des résultats théoriques d'une logique à une autre.
Dans [114], A. Salibra et G. Scollo établissent un tel résultat à propos de l'interpolation de Craig via la notion de transformation de pré-institutions, *i.e.* la propriété d'interpolation de Craig est vérifiée pour toute pré-institution source d'une transformation de pré-institutions dont la cible possède cette propriété. La notion de transformation que nous introduisons maintenant est une simple variante pour les institutions de la notion de transformation introduite dans [114].

Notation 2.3.1 *On introduit les notations suivantes :*

- *on note* $\wp^+ : \mathcal{Set} \to \mathcal{Set}$ *le foncteur qui à tout ensemble* $E \in |\mathcal{Set}|$ *associe la collection* $\wp(E) \subseteq |\mathcal{Set}|$ *privée de l'ensemble vide, i.e.* $\wp^+(E) = \wp(E)\backslash\{\varnothing\}$ *;*
- *pour toute institution* $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$*, on note* $\mathcal{Pres} = \wp \circ \mathcal{Sen}$ *le foncteur qui à toute signature* $\Sigma \in |\mathcal{Sig}|$ *associe la collection de tous les sous-ensembles* $E \subseteq \mathcal{Sen}(\Sigma)$*, i.e. toutes les* Σ*-présentations.*

Définition 2.3.1 (transformation) *Une* transformation d'institution $\tau : \mathcal{I} \to \mathcal{I}'$ *est la donnée d'un triplet* $(\mathcal{Si}, Pr, \mathcal{Mo})$ *tel que :*

- $\mathcal{Si} : \mathcal{Sig} \to \mathcal{Sig}'$ *est un* foncteur *;*
- $\mathcal{Pr} : \mathcal{Pres} \Rightarrow \mathcal{Pres}' \circ \mathcal{Si}$ *est une* transformation naturelle*, i.e. une famille de foncteurs indexée par* $|\mathcal{Sig}|$ *telle que pour tout morphisme de signatures* $\sigma \in \mathcal{Sig}(\Sigma_1, \Sigma_2)$ *le diagramme de la figure 2.4 commute.*

$$
\begin{tikzcd}
\Sigma_2 & \mathcal{Pres}(\Sigma_2) \arrow[r, "\mathcal{Pr}_{\Sigma_2}"] & \mathcal{Pres}'(\mathcal{Si}(\Sigma_2)) \\
\Sigma_1 \arrow[u, "\sigma"] & \mathcal{Pres}(\Sigma_1) \arrow[u, "\mathcal{Pres}(\sigma)"] \arrow[r, "\mathcal{Pr}_{\Sigma_1}"'] & \mathcal{Pres}'(\mathcal{Si}(\Sigma_1)) \arrow[u, "\mathcal{Pres}'(\mathcal{Si}(\sigma))"']
\end{tikzcd}
$$

FIG. 2.4 – *Translation de formules*

- $\mathcal{Mo} : \mathcal{Mod} \Rightarrow \wp^+ \circ \mathcal{Mod}' \circ \mathcal{Si}$ *est une* transformation naturelle*, i.e. une famille de foncteurs indexée par* $|\mathcal{Sig}|$ *telle que pour tout morphisme de signatures* $\sigma \in \mathcal{Sig}(\Sigma_1, \Sigma_2)$ *le diagramme de la figure 2.5 commute.*

$$
\begin{tikzcd}
\Sigma_2 & \mathcal{Mod}(\Sigma_2) \arrow[r, "\mathcal{Mo}_{\Sigma_2}"] \arrow[d, "\mathcal{Mod}(\sigma)"'] & \wp^+(\mathcal{Mod}'(\mathcal{Si}(\Sigma_2))) \arrow[d, "\wp^+(\mathcal{Mod}'(\mathcal{Si}(\sigma)))"] \\
\Sigma_1 \arrow[u, "\sigma"] & \mathcal{Mod}(\Sigma_1) \arrow[r, "\mathcal{Mo}_{\Sigma_1}"'] & \wp^+(\mathcal{Mod}'(\mathcal{Si}(\Sigma_1)))
\end{tikzcd}
$$

FIG. 2.5 – *Translation de modèles*

et vérifiant la propriété suivante :
$\forall \Sigma \in |\mathcal{Sig}|, \forall E \in \mathcal{Pres}(\Sigma), \forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,$

$$\mathcal{M} \models_\Sigma E \Longleftrightarrow (\forall \mathcal{M}' \in \mathcal{Mo}_\Sigma(\mathcal{M}),\ \mathcal{M}' \models'_{\Sigma'} \mathcal{Pr}_\Sigma(E))$$

Définition 2.3.2 (adéquate) *Une transformation d'institutions* $\tau : \mathcal{I} \to \mathcal{I}'$ *est dite* adéquate *si elle vérifie la propriété suivante :*
$\forall \Sigma \in |\mathcal{Sig}|, \forall E \in \mathcal{Pres}(\Sigma), \forall \mathcal{M}' \in |\mathcal{Mod}'(\mathcal{Si}(\Sigma))|,$

$$\mathcal{M}' \models'_{\mathcal{Si}(\Sigma)} \mathcal{Pr}_\Sigma(E) \Longrightarrow (\exists \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,\ \mathcal{M}' \in \mathcal{Mo}_\Sigma(\mathcal{M}) \wedge \mathcal{M} \models_\Sigma E)$$

Définition 2.3.3 (restriction adéquate) *Une transformation d'institutions* $\tau : \mathcal{I} \to \mathcal{I}'$ *est dite* restriction-adéquate *si elle vérifie la propriété suivante :*

$$\forall \sigma \in \mathcal{Sig}(\Sigma_1, \Sigma_2), \forall E \in \mathcal{Pres}(\Sigma_2),\ \mathcal{Sen}'(\mathcal{Si}(\sigma))(\mathcal{Pr}(E)) \subseteq \mathcal{Pr}(\mathcal{Sen}(\sigma)(E))$$

Théorème 2.3.1 ([114]) *Soit $\tau = (\mathcal{S}i, \mathcal{P}r, \mathcal{M}o) : \mathcal{I} \to \mathcal{I}'$ une transformation d'institutions adéquate et restriction-adéquate telle que $\mathcal{S}i$ préserve les pushouts. Alors, $\mathcal{I}$ possède la propriété d'interpolation de Craig sur les pushouts si $\mathcal{I}'$ la possède.*

2.3.2 Préservation par morphisme d'institutions

Nous nous intéressons maintenant à la préservation de l'interpolation de Craig lorsqu'on considère des restrictions d'un système logique donné. Pour cela, nous avons vu en section 1.8 que la notion adéquate de flèche dans ce cas est celle de morphisme d'institutions. Nous allons alors montrer que s'il existe un morphisme d'institution $\mu : \mathcal{I}' \to \mathcal{I}$ et que l'institution $\mathcal{I}'$ possède la propriété d'interpolation de Craig alors, sous certaines conditions très simples, l'institution $\mathcal{I}$ possède également cette propriété. Ce résultat est fortement basé sur le résultat de A. Salibra et G. Scollo présenté ci-dessus.

Théorème 2.3.2 *Soient $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ et $\mathcal{I}' = (\mathcal{S}ig', \mathcal{S}en', \mathcal{M}od', \models')$ deux institutions. Soit $\mu = (\Phi, \alpha, \beta) : \mathcal{I}' \to \mathcal{I}$ un morphisme d'institutions tel qu'il existe un foncteur $\overline{\Phi} : \mathcal{S}ig \to \mathcal{S}ig'$ vérifiant $\Phi \circ \overline{\Phi} = \mathtt{Id}_{\mathcal{S}ig}$ et que pour toute signature $\Sigma' \in |\mathcal{S}ig'|$, $\beta_{\Sigma'}$ et $\alpha_{\Sigma'}$ sont surjectifs. Alors, $\tau : \mathcal{I} \to \mathcal{I}'$ définie par le triplet $(\overline{\Phi}, \mathcal{P}r, \mathcal{M}o)$ tel que :*

- $\forall \Sigma \in |\mathcal{S}ig|, \forall E \subseteq \mathcal{S}en(\Sigma)$, $\mathcal{P}r_\Sigma(E) = \{\alpha_{\overline{\Phi}(\Sigma)}(\varphi) / \varphi \in E\}$;
- $\forall \Sigma \in |\mathcal{S}ig|$, $\mathcal{M}o_\Sigma : \begin{array}{rcl} Mod(\Sigma) & \to & \wp^+(Mod'(\overline{\Phi}(\Sigma))) \\ \mathcal{M} & \mapsto & \beta^{-1}_{\overline{\Phi}(\Sigma)}(\mathcal{M}) \end{array}$

est une transformation d'institution adéquate et restriction adéquate.

Preuve

- La naturalité de $\mathcal{P}r$ and $\mathcal{M}o$ résulte de celles de α et β.
- Soient $\Sigma \in |\mathcal{S}ig|$, $E \subseteq \mathcal{S}en(\Sigma)$ et $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ tels que $\mathcal{M} \models_\Sigma E$. Soit $\mathcal{M}' \in |\mathcal{M}od'(\overline{\Phi}(\Sigma))|$ tel que $\beta_{\overline{\Phi}(\Sigma)}(\mathcal{M}') = \mathcal{M}$. Par la propriété de satisfaction des morphismes d'institutions, $\mathcal{M}' \models'_{\overline{\Phi}(\Sigma)} \mathcal{P}r_\Sigma(E)$.
- Pour montrer l'adéquation, considérons maintenant un ensemble $E \subseteq \mathcal{S}en(\Sigma)$ de Σ-formules et un modèle $\mathcal{M}' \in |\mathcal{M}od'(\overline{\Phi}(\Sigma))|$ tels que $\mathcal{M}' \models'_{\overline{\Phi}(\Sigma)} \mathcal{P}r_\Sigma(E)$. Par la propriété de satisfaction des morphismes d'institutions, $\beta_{\overline{\Phi}(\Sigma)}(\mathcal{M}') \models_\Sigma E$.
- Finalement, pour montrer la restriction-adéquation, considérons un morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ dans $\mathcal{I}$ et un ensemble $E \subseteq \mathcal{S}en(\Sigma_2)$ de formules. Soit $\psi' \in \mathcal{P}r_{\Sigma_1}(E_\sigma)$, où $E_\sigma = \{\varphi \in \mathcal{S}en(\Sigma_1) / E \models_{\Sigma_2} \mathcal{S}en(\sigma)(\varphi)\}$. Alors, ψ' est de la forme $\alpha_{\overline{\Phi}(\Sigma_1)}(\varphi)$, où $\varphi \in \mathcal{S}en(\Sigma_1)$ telle que $E \models_{\Sigma_2} \mathcal{S}en(\sigma)(\varphi)$. Montrons que $\mathcal{P}r_{\Sigma_2}(E) \models_{\overline{\Phi}(\Sigma_2)} \alpha_{\overline{\Phi}(\Sigma_2)}(\mathcal{S}en(\sigma)(\varphi))$. Soit $\mathcal{M}' \models_{\overline{\Phi}(\Sigma_2)} \mathcal{P}r_{\tau_{\Sigma_2}}(E)$. Par la propriété de satisfaction des morphismes d'institutions, $\beta_{\overline{\Phi}(\Sigma_2)}(\mathcal{M}') \models_{\Sigma_2} E$. Ainsi, $\beta_{\overline{\Phi}(\Sigma_2)}(\mathcal{M}') \models_{\Sigma_2} \mathcal{S}en(\sigma)(\varphi)$. Par la propriété de satisfaction des morphismes d'institutions, on en déduit $\mathcal{M}' \models_{\overline{\Phi}(\Sigma_2)} \psi$.

□

L'hypothèse de surjectivité de β_Σ est assez naturelle, de même que l'existence de $\overline{\Phi}$. En revanche, la surjectivité de α_Σ est souvent trop forte. En effet, de nombreux morphismes de logiques ne satisfont pas cette propriété. C'est le cas par exemple du morphisme entre la logique du premier ordre avec égalité et sans prédicat et la logique équationnelle. Nous définissons ci-dessous une condition plus faible bien qu'également non vérifiée par le morphisme que l'on vient d'évoquer.

Définition 2.3.4 *Soit $\mu : \mathcal{I}' \to \mathcal{I}$ un morphisme d'institutions tel que $\beta_{\Sigma'}$ est surjectif pour toute signature $\Sigma' \in |\mathcal{Sig}'|$. On dit que le morphisme μ est* conservatif sur les formules *par rapport au foncteur $\overline{\Phi} : \mathcal{Sig} \to \mathcal{Sig}'$ tel que $\Phi \circ \overline{\Phi} = \mathtt{Id}_{\mathcal{Sig}}$ si :*
$\forall \Sigma \in |\mathcal{Sig}|, \forall \varphi' \in \mathcal{Sen}'(\overline{\Phi}(\Sigma)) \setminus \alpha_{\overline{\Phi}(\Sigma)}(\mathcal{Sen}(\Sigma)), \exists E \subseteq \mathcal{Sen}(\Sigma),$

$$\mathcal{Mod}'(\{\varphi'\}) = \mathcal{Mod}'(\alpha_{\overline{\Phi}(\Sigma)}(E))$$

Théorème 2.3.3 *Soient $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ et $\mathcal{I}' = (\mathcal{Sig}', \mathcal{Sen}', \mathcal{Mod}', \models')$ deux institutions et soit $\mu = (\Phi, \alpha, \beta) : \mathcal{I}' \to \mathcal{I}$ un morphisme d'institutions conservatif sur les formules. Alors, si $\mathcal{I}'$ possède la propriété d'interpolation de Craig pour une classe C' de diagramme, $\mathcal{I}$ la possède également pour la classe $\overline{\Phi}(C')$.*

Preuve Soit

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle \sigma_2}\downarrow & & \downarrow{\scriptstyle \sigma_1^\sharp} \\ \Sigma_2 & \xrightarrow[\sigma_2^\sharp]{} & \Sigma^\sharp \end{array}$$

un diagramme commutatif de morphismes de signatures dans $\mathcal{Sig}$. Soient $E_1 \subseteq \mathcal{Sen}(\Sigma_1)$ et $E_2 \subseteq \mathcal{Sen}(\Sigma_2)$ des ensembles de $\mathcal{I}$-formules tels que $\mathcal{Sen}(\sigma_1)(E_1) \models_{\Sigma^\sharp} \mathcal{Sen}(\sigma_2)(E_2)$. Soit $\mathcal{M}' \in \mathcal{Mod}'(\alpha_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{Sen}(\sigma_1)(E_1)))$. Puisque μ est un morphisme d'institutions, par la propriété de satisfaction, on a :

$$\begin{array}{rcl} \mathcal{M}' \models'_{\overline{\Phi}(\Sigma^\sharp)} \alpha_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{Sen}(\sigma_1)(E_1)) & \Leftrightarrow & \beta_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{M}') \models'_{\Sigma^\sharp} \mathcal{Sen}(\sigma_1)(E_1) \\ & \Rightarrow & \beta_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{M}') \models'_{\Sigma^\sharp} \mathcal{Sen}(\sigma_2)(E_2) \\ & \Leftrightarrow & \mathcal{M}' \models'_{\overline{\Phi}(\Sigma^\sharp)} \alpha_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{Sen}(\sigma_2)(E_2)) \end{array}$$

On a donc $\alpha_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{Sen}(\sigma_1)(E_1)) \models_{\overline{\Phi}(\Sigma^\sharp)} \alpha_{\overline{\Phi}(\Sigma^\sharp)}(\mathcal{Sen}(\sigma_2)(E_2))$. Puisque $\mathcal{I}'$ possède la propriété d'interpolation, il existe $E' \subseteq \mathcal{Sen}'(\overline{\Phi}(\Sigma))$ tel que $\alpha_{\overline{\Phi}(\Sigma_1)}(E_1) \models'_{\Sigma_1} \mathcal{Sen}'(\overline{\Phi}(\sigma_1))(E')$ et $\mathcal{Sen}'(\overline{\Phi}(\sigma_2))(E'_2) \models'_{\Sigma_2} \alpha_{\overline{\Phi}(\Sigma_2)}(E_2)$. Puisque μ est conservatif pour les formules, il existe $E \subseteq \mathcal{Sen}(\Sigma)$ tel que $\mathcal{Mod}'(E) = \mathcal{Mod}'(\alpha_{\overline{\Phi}(\Sigma)}(E'))$. Soit $\mathcal{M} \in \mathcal{Mod}(E_1)$. Puisque $\beta_{\overline{\Phi}(\Sigma_1)}$ est surjectif, il existe $\mathcal{M}' \in |\mathcal{Mod}'(\overline{\Phi}(\Sigma_1))|$ tel que $\beta_{\overline{\Phi}(\Sigma_1)}(\mathcal{M}') = \mathcal{M}$. Maintenant, par la propriété de satisfaction des morphismes d'institutions, on a :

$$\begin{array}{rcl} \mathcal{M} \models_{\Sigma_1} E_1 & \Leftrightarrow & \mathcal{M}' \models'_{\overline{\Phi}(\Sigma_1)} \alpha_{\overline{\Phi}(\Sigma_1)}(E_1) \\ & \Rightarrow & \mathcal{M}' \models'_{\overline{\Phi}(\Sigma_1)} \mathcal{Sen}'(\overline{\Phi}(\sigma_1))(\alpha_{\overline{\Phi}(\Sigma)}(E)) \\ & \Leftrightarrow & \mathcal{M} \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(E) \end{array}$$

On a ainsi montré $E_1 \models_{\Sigma_1} \mathcal{Sen}(\sigma_1)(E)$. On montre de même que $\mathcal{Sen}(\sigma_2)(E) \models_{\Sigma_2} E_2$. $\square$

Chapitre 3

Théorème de Beth

Lorsque l'on désire formaliser une théorie, la première chose à faire est de fixer le langage, *i.e.* décider quelles seront les notions primitives et celles qui devront être définies à partir de ces notions premières. L'une des questions qui se pose alors est de savoir si certains symboles sont superflus, *i.e.* s'il n'est pas possible de les exprimer à partir des notions primitives. Pour modéliser l'arithmétique par exemple, trois symboles uniquement sont nécessaires (un symbole de constante pour le 0, un symbole de fonction unaire pour la fonction successeur et un symbole de fonction binaire pour l'addition). Toutes les autres notions de l'arithmétique (multiplication, relation d'ordre, nombres premiers, *etc.*) pouvant être définies à partir de ces trois symboles. Le théorème de Beth (*cf.* [15]), initialement démontré par E. Beth pour la logique du 1^{er}ordre classique, offre un critère sémantique permettant de répondre à cette question. Dans le cadre de la théorie standard des modèles, ce résultat est une conséquence du théorème d'interpolation de Craig.

Dans ce chapitre, nous étudions le problème de la définissabilité dans un cadre institution-indépendant. Nous abordons cette généralisation sous deux angles différents :

1. nous définissons tout d'abord la définissabilité dans le cadre restreint de l'inclusion de signatures. Nous montrons alors que l'équivalence obtenue par E. Beth dans le cadre de la théorie standard des modèles n'est pas vérifiée dans le cadre abstrait des institutions. Seul le sens « implicite » vers « explicite » est conservée ;
2. nous généralisons ensuite la notion de définissabilité à tout type de morphisme de signatures. Ceci nous permet alors de retrouver l'équivalence entre définissabilité implicite et explicite. Ainsi, le problème classique de définissabilité d'un nouveau symbole χ par rapport à une signature donnée Σ (déterminant une inclusion de signature $\Sigma \hookrightarrow \Sigma \cup \{\chi\}$) est généralisé et abstrait à tout morphisme de signatures $\sigma : \Sigma \to \Sigma'$ d'une institution quelconque. En accord avec [41], nous pensons que ceci est le vrai concept (abstrait) de définissabilité.

Au niveau abstrait des institutions, les signatures sont simplement des objets d'une catégorie. De façon à dénoter des symboles pouvant être définis à partir d'autres, il nous faut définir la notion d'inclusion de signatures. Ceci demande de définir la notion ensembliste d'inclusion en termes catégoriels. Pour ce faire, nous utilisons la notion de *catégorie fortement inclusive* introduite dans [45].

3.1 Définitions catégorielles d'opérations ensemblistes

La notion de catégorie fortement inclusive introduite dans [45] permet de donner une définition catégorielle des notions ensemblistes d'inclusion, d'union et d'intersection. La définition catégorielle de la différence ensembliste que nous présentons ici a été précédemment proposées dans [5].

Définition 3.1.1 (système d'inclusion) *Une catégorie $\mathcal{C}$ est dite* fortement inclusive *si et seulement si elle contient des pullbacks*[1] *et s'il existe deux sous-catégories $\mathcal{I}$ et $\mathcal{E}$ telles que :*

- $|\mathcal{I}| = |\mathcal{E}| = |\mathcal{C}|$;
- *tout morphisme $e \in \mathcal{E}$ est un épimorphisme*[2] ;
- *pour tout morphisme $f \in \mathcal{C}$ il existe un unique couple (ι, e) de morphisme de $\mathcal{C}$ tel que :*
  - $\iota \in \mathcal{I}$,
  - $e \in \mathcal{E}$,
  - $f = \iota \circ e$.
- *$\mathcal{I}$ définit un ordre partiel sur $\mathcal{C}$ et l'ensemble partiellement ordonné $(|\mathcal{C}|, \mathcal{I})$ est un treillis pour lequel le sup de tout couple d'objets (a, b) est la somme de a et b, notée $a + b$, et l'inf de a et b est l'unique pullback d'inclusion de la somme $a + b$ dans $\mathcal{C}$, appellé* intersection *de a et b et notée $a \cap b$*[3]. *De plus, nous imposons que $a + b$ soit le pushout de $a \cap b$ dans $\mathcal{C}$.*

Les morphismes de $\mathcal{I}$ sont appelés *inclusions* et le couple $(\mathcal{I}, \mathcal{E})$ est appelé *système d'inclusion* de $\mathcal{C}$. Nous utiliserons désormais la notation $\iota : a \hookrightarrow b$ pour indiquer que le morphisme $\iota : a \to b$ est un morphisme de $\mathcal{I}(a, b)$.

Remarque 3.1.1 *Pour toute catégorie $\mathcal{C}$, la sous-catégorie de $\mathcal{C}$ dont les objets sont ceux de $\mathcal{C}$ et dont les morphismes sont les inclusions est* finiment cocomplète (*cf.* [45] pour la preuve).

Exemple 3.1.1 *La catégorie $\mathcal{S}et$ munie du système d'inclusion $(\mathcal{I}, \mathcal{E})$, où $\mathcal{I}$ contient toutes les inclusions et $\mathcal{E}$ toutes les surjections, est une catégorie fortement inclusive.*
En effet, soit le pushout de la figure 3.1 dans $\mathcal{S}et$, où $E_0 = E_1 \cap E_2$ et dont toutes les flèches sont des inclusions. L'ensemble E est alors l'union ensembliste de E_1 et E_2, i.e. $E = E_1 \cup E_2$ et puisque ce pushout est également un pullback, on a $E_1 \cup E_2$ est le pushout de $E_1 \cap E_2$.

Proposition 3.1.1 ([45]) *Soit $\mathcal{C}$ une catégorie fortement inclusive et $(\mathcal{I}, \mathcal{E})$ son système d'inclusion. Si $\mathcal{C}$ possède un objet initial, noté $\emptyset$, alors $\emptyset$ est également initial dans $\mathcal{I}$.*

1. *cf.* définition en annexe A
2. *Idem.*
3. Les conditions d'existence et d'unicité de $a \cap b$ sont données dans [45]

$$\begin{tikzcd} E_0 \arrow[r, hook] \arrow[d, hook] & E_1 \arrow[d, hook] \\ E_2 \arrow[r, hook] & E \end{tikzcd}$$

FIG. 3.1 – *Pushout d'inclusion dans $\mathcal{S}et$*

L'inclusion ensembliste est l'exemple le plus simple et le plus intuitif de systèmes d'inclusion. Cependant, il faut prendre garde au fait que certaines propriétés de l'inclusion ensembliste découlent de la définition des ensembles et non de celle des systèmes d'inclusion (*cf.* [45]). Par exemple, toute inclusion $A \hookrightarrow B$ en théorie des ensembles est « coupée », au sens où B peut être écrit comme l'union disjointe $A \coprod C$ où C est la « différence » entre B et A. Cette dernière propriété n'est pas vraie en général pour les systèmes d'inclusion. Ainsi, dans le cas d'une inclusion de signatures équtionnelles multi-sortes $(\{s\}, \emptyset) \hookrightarrow (\{s\}, \{f : s \to s\})$, la difference $(\emptyset, \{f : s \to s\})$ ensembliste n'est pas une signature[4]. Néanmoins, cette notion peut être exprimée de façon catégorielle si on ne la base pas sur les propriétés de l'union disjointe. En effet, il est possible de caractériser la différence entre deux ensembles B et A, notée $B \setminus A$, comme le plus grand sous-ensemble X de B tel que $X \cap A = \emptyset$.
Dans la suite, nous supposerons donc que toutes catégorie fortement inclusive $\mathcal{C}$ vérifie les conditions supplémentaires suivantes :

Définition 3.1.2 (différence catégorielle) *Soit $\mathcal{C}$ une catégorie fortement inclusive et soit $\emptyset$ son objet initial. La catégorie $\mathcal{C}$ satisfait la* propriété de différence *si pour tout objet $c \in |\mathcal{C}|$ il existe un foncteur $\_ \setminus c : \mathcal{C} \to \mathcal{C}$ tel que pour tout objet c', l'objet $c' \setminus c$ est l'objet terminal de la sous-catégorie pleine de $\mathcal{C}$ dont la classe d'objets est la classe $\{X \mid X \cap c = \emptyset \wedge X \hookrightarrow c'\}$.*
Étant donnés deux objets $c, c' \in |\mathcal{C}|$, l'objet $c' \setminus c$ est appelé différence *de c' et c.*

Notation 3.1.1 *Soit $\mathcal{C}$ une catégorie et soient $c, c' \in |\mathcal{C}|$ deux objets de $\mathcal{C}$. Nous noterons $c \approx c'$ le fait que c et c' sont deux objets isomorphiques, i.e. il existe deux morphismes $\alpha : c \to c'$ et $\alpha^{-1} : c' \to c$ tels que $\alpha \circ \alpha^{-1} = Id_{c'}$ et $\alpha^{-1} \circ \alpha = Id_c$.*

Proposition 3.1.2 ([5]) *Soit $\mathcal{C}$ une catégorie fortement inclusive et soient $c, c_1, c_2 \in |\mathcal{C}|$ des objets de $\mathcal{C}$. Si $c \hookrightarrow c_i$ ($i \in \{1, 2\}$) et $c_1 \approx c_2$ alors $c_1 \setminus c \approx c_2 \setminus c$.*

3.2 Formulation institution-indépendante restreinte

La notion de *définissabilité* dénote le fait que certaines notions sont exprimables à partir de notions pré-existantes, *i.e.* elles offrent un confort d'écriture mais ne permettent pas d'exprimer plus de concepts qu'auparavant. Les qualificatifs *explicitement* et *implicitement* traduisent respectivement le fait qu'une notion soit définie syntaxiquement (*i.e.* au moyen de formules) ou

4. Un résultat correct pour cette différence aurait été $(\varnothing, \varnothing)$

sémantiquement (*i.e.* au moyen de modèles).

Le théorème de Beth établit l'équivalence, dans le cadre de la logique du premier ordre classique, entre les notions de définissabilité explicite et implicite. Pour généraliser ces deux notions dans le langage des institutions, il nous faut considérer des institutions semi-exactes, fermées par négation et satisfaisant les deux conditions suivantes :

1. $\mathcal{S}ig$ est une catégorie fortement inclusive. On note $(\mathcal{I}, \mathcal{E})$ son système d'inclusion. Elle possède un objet initial noté $\Sigma_\emptyset$ qui est également initial dans $\mathcal{I}$. De plus, $\mathcal{S}ig$ satisfait la propriété de différence catégorielle ;
2. $\forall(\Sigma \hookrightarrow \Sigma') \in \mathcal{I}$, $\mathcal{S}en(\Sigma) \subseteq \mathcal{S}en(\Sigma')$.

Définition 3.2.1 (définissabilité implicite) *Soient $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et $(\mathcal{I}, \mathcal{E})$ le système d'inclusion associé à $\mathcal{S}ig$. Soient $\Sigma \hookrightarrow \Sigma' \in \mathcal{I}$ et $T \in \mathcal{T}h(\Sigma')$ une Σ'-théorie. $\Sigma' \setminus \Sigma$ est* implicitement définissable par T *si pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ et pour tout Σ'-modèles $\mathcal{M}', \mathcal{M}'' \in |\mathcal{M}od(T)|$ tels que $\mathcal{M}od(\Sigma \hookrightarrow \Sigma')(\mathcal{M}') = Mod(\Sigma \hookrightarrow \Sigma')(\mathcal{M}'') = \mathcal{M}$, on a $\mathcal{M}' \equiv_{\Sigma'} \mathcal{M}''$.*

Notons que la notion de définissabilité implicite que nous venons de définir est plus libérale que celle considérée en théorie standard des modèles. Ceci vient du fait que nous nous contentons de l'équivalence élémentaire des T-modèles $\mathcal{M}'$ et $\mathcal{M}''$ alors que la notion usuelle de définissabilité implicite impose leur égalité.

Définition 3.2.2 (définissabilité explicite) *Soient $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et $(\mathcal{I}, \mathcal{E})$ le système d'inclusion associé à $\mathcal{S}ig$. Soit $\Sigma \hookrightarrow \Sigma' \in \mathcal{I}$ et $T \in \mathcal{T}h(\Sigma')$ une Σ'-théorie. $\Sigma' \setminus \Sigma$ est* explicitement définissable par rapport à T *si pour toute Σ'-formule $\varphi \in \mathcal{S}en(\Sigma' \setminus \Sigma)$ il existe un ensemble de Σ-formules $E \subseteq \mathcal{S}en(\Sigma)$ tel que les deux conditions suivantes soient vérifiées :*

1. $T \cup \{\varphi\} \models_{\Sigma'} E$
2. $T \cup E \models_{\Sigma'} \varphi$

Du fait de la structure inductive des formules de la logique du premier ordre classique, il est clair que, dans le cadre de la théorie standard des modèles, la définissabilité implicite est une conséquence de la définissabilité explicite. En effet, par compacité on peut montrer que pour toute formule $\varphi \in \mathcal{S}en(\Sigma' \setminus \Sigma)$ (où Σ' et Σ sont des signatures du 1^{er}ordre), l'ensemble E de la définition 3.2.2 est restreint à une Σ-formule $\psi_\varphi \in \mathcal{S}en(\Sigma)$. On peut donc, pour toute Σ'-formule $\Phi \in \mathcal{S}en(\Sigma')$, construire une Σ-formule $\Psi \in \mathcal{S}en(\Sigma)$ telle que $\mathcal{M}od(T \cup \{\Phi\}) = \mathcal{M}od(T \cup \{\Psi\})$. Cette Σ-formule Ψ est obtenue en remplaçant les occurences de toute Σ'-formule atomique $p'(t_1, \ldots, t_n) \in \mathcal{S}en(\Sigma' \setminus \Sigma)$ par $\psi_{p'(t_1,\ldots,t_n)} \in \mathcal{S}en(\Sigma)$. Alors, pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ et tout modèle $\mathcal{M}', \mathcal{M}'' \in \mathcal{M}od(T)$ tels que $\mathcal{M}od(\Sigma \hookrightarrow \Sigma')(\mathcal{M}') = \mathcal{M}od(\Sigma \hookrightarrow \Sigma')(\mathcal{M}'') = \mathcal{M}$, on a $\mathcal{M}' \equiv_{\Sigma'} \mathcal{M}''$. On a en effet :

$$\begin{aligned}
\mathcal{M}' \models_{\Sigma'} \Phi &\iff \mathcal{M}' \models_{\Sigma'} \Psi \\
&\iff Mod(\Sigma \hookrightarrow \Sigma')(\mathcal{M}') \models_{\Sigma} \Psi \\
&\iff \mathcal{M}'' \models_{\Sigma'} \Psi \\
&\iff \mathcal{M}'' \models_{\Sigma'} \Phi
\end{aligned}$$

La *propriété de Beth* caractérise les logiques pour lesquelles l'implication inverse est vérifiée. C'est le résultat que E. Beth a montré en 1953 pour la logique du premier ordre classique (*cf.* [15]).

Définition 3.2.3 (propriété de Beth) *Une institution $\mathcal{I}$ possède la* propriété de Beth *si pour tout $\Sigma \hookrightarrow \Sigma'$ et toute Σ'-théorie T, si $\Sigma' \setminus \Sigma$ est implicitement définissable par T alors $\Sigma' \setminus \Sigma$ est explicitement définissable par rapport à T. Dans ce cas, nous dirons que $\Sigma' \setminus \Sigma$ possède la propriété de Beth pour T.*

Dans le cadre des institutions nous devons, de façon à assurer que la version explicite de la définissabilité implique la version implicite, étendre la notion de définissabilité explicite en remplaçant dans la définition 3.2.2 la phrase « pour toute Σ'-formule $\varphi \in \mathcal{Sen}(\Sigma' \setminus \Sigma)$ » par la phrase « pour toute Σ'-formule $\varphi \in \mathcal{Sen}(\Sigma') \setminus \mathcal{Sen}(\Sigma)$ ». Nous obtenons alors le résultat suivant :

Proposition 3.2.1 *Si $\Sigma' \setminus \Sigma$ est explicitement définissable par rapport à T alors elle est implicitement définissable pour T.*

Preuve Soit $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$ et $\mathcal{M}_i \in \mathcal{Mod}(T)$ tels que $\mathcal{Mod}(\Sigma \hookrightarrow \Sigma')(\mathcal{M}_i) = \mathcal{M}$ $(i = 1, 2)$. Soit $\varphi \in \mathcal{Sen}(\Sigma')$ telle que $\mathcal{M}_i \models_{\Sigma'} \varphi$. Montrons que $\mathcal{M}_j \models_{\Sigma'} \varphi$ $(j \neq i$ et $i, j \in \{1, 2\})$. Par la condition de satisfaction, cela est évident si $\varphi \in \mathcal{Sen}(\Sigma)$. Soit alors $\varphi \in \mathcal{Sen}(\Sigma') \setminus \mathcal{Sen}(\Sigma)$. Puisque $\Sigma' \setminus \Sigma$ est explicitement définissable par rapport à T, il existe un ensemble E tel que $T \cup E \models_{\Sigma'} \varphi$ and $T \cup \{\varphi\} \models_{\Sigma'} E$. Supposons que $\mathcal{M}_j \not\models_{\Sigma'} \varphi$. Puisque $\mathcal{I}$ est fermée par négation, on a $\mathcal{M}_j \models_{\Sigma'} \neg\varphi$. Alors, il existe une formule $\psi \in E$ telle que $\mathcal{M}_j \models_{\Sigma'} \neg\psi$. Par la condition de satisfaction, nous avons $\mathcal{M} \models_{\Sigma} \mathcal{Sen}(\Sigma \hookrightarrow \Sigma')(\neg\psi)$. Or, puisque $T \cup \{\varphi\} \models_{\Sigma'} E$, nous avons également $\mathcal{M} \models_{\Sigma'} Sen(\Sigma \hookrightarrow \Sigma')(\psi)$ ce qui est impossible. □

Cette nouvelle notion de définissabilité explicite n'est pas une conséquence de celle donnée par la définition 3.2.2. La raison en est que dans les institutions les formules sont uniquement définies comme éléments d'un ensemble. Cependant, la preuve donnée ci-dessus pour la logique du premier ordre nous laisse supposer qu'un tel résultat serait valide dans les cadres abstraits où les formules répondent à une définition inductive (typiquement les différentes versions de parchemins [*cf.* [62, 98, 122]]). De plus, même si en théorie standard des modèles la propriété de Beth est définie pour des formules atomiques, cela est suffisant pour étendre cette propriété à une notion plus générale d'inclusion de signatures. En particulier, la logique du premier ordre vérifie les résultats suivants :

Proposition 3.2.2 *En logique du premier ordre, si $\Sigma' \setminus \Sigma$ et $\Sigma'' \setminus \Sigma$ sont explicitement définissable (au nouveau sens) pour T' et T'' respectivement, alors $\Sigma' + \Sigma'' \setminus \Sigma$ est explicitement définissable pour $\mathcal{Th}(T' \cup T'')$.*

Preuve Soit $\varphi \in \mathcal{Sen}(\Sigma' + \Sigma'') \setminus \mathcal{Sen}(\Sigma)$. Montrons qu'il existe $E \subseteq \mathcal{Sen}(\Sigma)$ tel que :

1. $\mathcal{Th}(T' \cup T'') \cup \varphi \models_{\Sigma'+\Sigma''} E$;
2. $\mathcal{Th}(T' \cup T'') \cup E \models_{\Sigma'+\Sigma''} \varphi$.

La démonstration se fait par induction sur la structure de φ.

cas de base : φ est de la forme $p(t_1, \ldots, t_n)$. Soit $p \in \Sigma'$, *i.e.* φ est une Σ'-formule. Puisque $\Sigma' \setminus \Sigma$ est explicitement définissable pour T', il existe un ensemble $E \subseteq \mathcal{Sen}(\Sigma)$ tel que :

1. $T' \cup \{\varphi\} \models_{\Sigma'} E$;

2. $T' \cup E \models_{\Sigma'} \varphi$.

cas général : par la proposition 1.3.2, on a :

1. $T' \cup \{\varphi\} \models_{\Sigma'+\Sigma''} E$;
2. $T' \cup E \models_{\Sigma'+\Sigma''} \varphi$.

Or, $\mathcal{Th}(T' \cup T'') \models_{\Sigma'+\Sigma''} T'$, on a alors le résultat voulu. On prouve le cas $p \in \Sigma''$ de façon similaire.

□

Ce dernier résultat n'est pas valable pour la définissabilité implicite lorsqu'on impose aux structures du premier ordre $\mathcal{M}_1$ et $\mathcal{M}_2$ de n'être qu'élémentairement équivalentes. Cependant, si on se restreint au cadre de la théorie standard des modèles, *i.e.* la définissabilité implicite repose sur l'égalité de $\mathcal{M}_1$ et $\mathcal{M}_2$, alors le résultat ci-dessus est valable lorsque l'on remplace « explicite » par « implicite ».

3.3 Preuve institution-independante

On généralise maintenant dans un cadre institution-indépendant le résultat démontré par E. Beth, *i.e.* l'équivalence des notions de définissabilité implicites et explicites sous l'hypothèse de l'interpolation de Craig.

Lemme 3.3.1 *Soit $\Sigma' \setminus \Sigma$ une signature implicitement définissable pour une Σ'-theorie T. Soit $\Sigma \hookrightarrow \Sigma''$ une inclusion de signatures telle que Σ' est isomorphe à Σ'' (i.e. il existe un morphisme de signatures $\alpha : \Sigma' \to \Sigma''$ tel que $\alpha^{-1} \circ \alpha = \mathtt{Id}_{\Sigma'}$ et $\alpha \circ \alpha^{-1} = \mathtt{Id}_{\Sigma''}$). Alors, pour tout ensemble de formules $E \subseteq \mathcal{Sen}(\Sigma' \setminus \Sigma)$, on a :*

$$T \cup E \models_{\Sigma'+\Sigma''} \mathcal{Sen}(\alpha)(T \cup E)$$

Preuve Supposons $\mathcal{M} \models_{\Sigma'+\Sigma''} T \cup E$. Par la proposition 1.3.4, on a alors $\mathcal{Mod}(\Sigma' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M}) \models_{\Sigma'} \mathcal{Sen}(\alpha^{-1} \circ \alpha)(T \cup E)$ et donc $\mathcal{Mod}(\alpha^{-1})(\mathcal{Mod}(\Sigma' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M})) \models_{\Sigma''} \mathcal{Sen}(\alpha)(T \cup E)$. Bien entendu, on a : $(\Sigma'' \hookrightarrow \Sigma' + \Sigma'') \circ \alpha \circ (\Sigma \hookrightarrow \Sigma') = (\Sigma' \hookrightarrow \Sigma' + \Sigma'') \circ (\Sigma \hookrightarrow \Sigma')$. On a alors :

$$\begin{gathered} \mathcal{Mod}(\Sigma \hookrightarrow \Sigma')(\mathcal{Mod}(\alpha)(\mathcal{Mod}(\Sigma'' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M}))) \\ = \\ \mathcal{Mod}(\Sigma \hookrightarrow \Sigma')(\mathcal{Mod}(\Sigma' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M})) \end{gathered}$$

Puisque $\Sigma' \setminus \Sigma$ est implicitement définissable pour T, on a :

$$\mathcal{Mod}(\alpha)(\mathcal{Mod}(\Sigma'' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M})) \equiv_{\Sigma'} \mathcal{Mod}(\Sigma' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M})$$

On a alors :

$$\mathcal{Mod}(\alpha^{-1})(\mathcal{Mod}(\alpha)(\mathcal{Mod}(\Sigma'' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M}))) \models_{\Sigma''} \mathcal{Sen}(\alpha)(T \cup E)$$

i.e. $\mathcal{Mod}(\Sigma'' \hookrightarrow \Sigma' + \Sigma'')(\mathcal{M}))) \models_{\Sigma''} \mathcal{Sen}(\alpha)(T \cup E)$. Par la condition de satisfaction, on en déduit : $\mathcal{M} \models_{\Sigma'+\Sigma''} \mathcal{Sen}(\alpha)(T \cup E)$. □

Théorème 3.3.1 *Soit $\mathcal{I}$ une institution vérifiant l'interpolation de Craig. Alors, $\mathcal{I}$ possède la propriété de Beth.*

Preuve Soit $\Sigma' \setminus \Sigma$ une signature implicitement définissable pour une Σ'-théorie T et soit $\varphi \in \mathcal{Sen}(\Sigma' \setminus \Sigma)$. Soit $\Sigma' \hookrightarrow \Sigma''$ une inclusion de signatures telle que Σ' et Σ'' sont isomorphes. Par le lemme 3.3.1, on a $T \cup \{\varphi\} \models_{\Sigma'+\Sigma''} \mathcal{Sen}(\alpha)(\varphi)$. Par le théorème d'interpolation de Craig, il existe $E' \subseteq \mathcal{Sen}(\Sigma)$ tel que $T \cup \{\varphi\} \models_{\Sigma'} E'$ et $E' \models_{\Sigma''} \mathcal{Sen}(\alpha)(\varphi)$. Soit $\mathcal{M} \in |\mathcal{Mod}(\Sigma')|$ tel que $\mathcal{M} \models_{\Sigma'} E'$. Par la condition de satisfaction on a $\mathcal{Mod}(\alpha^{-1})(\mathcal{M}) \models_{\Sigma''} E'$. Alors, $\mathcal{Mod}(\alpha^{-1})(\mathcal{M}) \models_{\Sigma''} \mathcal{Sen}(\alpha)(\varphi)$. Par la condition de satisfaction, de nouveau, on a $\mathcal{M} \models_{\Sigma'} \varphi$. Alors, $E' \models_{\Sigma'} \varphi$ et donc $T \cup E' \models_{\Sigma'} \varphi$. □

Notons que le théorème précédent ne suppose pas d'autres hypothèses que la propriété d'interpolation de Craig. Puisque, en particulier, on ne suppose pas la négation, ceci est également vrai pour les institutions de Birkhoff possédant l'interpolation de Craig.

3.4 Préservation de la propriété de Beth

De même que pour l'interpolation de Craig, nous étudions maintenant les conditions de transfert de la propriété de Beth au travers des morphismes d'institutions. Ceci demande de considérer deux conditions supplémentaires à ajouter à celle donnée dans le théorème 2.3.3.

Définition 3.4.1 (conservatif) *Soient deux institutions $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ et $\mathcal{I}' = (\mathcal{Sig}', \mathcal{Sen}', \mathcal{Mod}', \models')$ et soit $\mu = (\Phi, \alpha, \beta) : \mathcal{I} \to \mathcal{I}'$ un morphisme d'institutions tel que β_Σ soit surjectif. Le morphisme μ est dit* conservatif par rapport à un foncteur $\overline{\Phi} : Sig' \to Sig$ si et seulement si :

1. pour tout $\sigma' : \Sigma'_1 \to \Sigma'_2$ de $\mathcal{I}'$, pour tout $\mathcal{M}'_1, \mathcal{M}'_2 \in |\mathcal{Mod}'(\Sigma'_2)|$:

$$\mathcal{Mod}'(\sigma')(\mathcal{M}'_1) = \mathcal{Mod}'(\sigma')(\mathcal{M}'_2)$$
$$\Longrightarrow$$
$$\exists \mathcal{M}_1 \in \beta^{-1}_{\overline{\Phi}(\Sigma'_2)}(\mathcal{M}'_1), \exists \mathcal{M}_2 \in \beta^{-1}_{\overline{\Phi}(\Sigma'_2)}(\mathcal{M}'_2),$$
$$\mathcal{Mod}(\overline{\Phi}(\sigma'))(\mathcal{M}_1) = \mathcal{Mod}(\overline{\Phi}(\sigma'))(\mathcal{M}_2)$$

2. pour toute signature $\Sigma' \in |Sig'|$, pour toute formule $\varphi \in \mathcal{Sen}(\overline{\Phi}(\Sigma')) \setminus \alpha_{\overline{\Phi}(\Sigma')}(\mathcal{Sen}'(\Sigma'))$ il existe $E' \subseteq \mathcal{Sen}'(\Sigma')$ tel que $\mathcal{Mod}(\varphi) = \mathcal{Mod}(\alpha_{\overline{\Phi}(\Sigma')}(E'))$.

Théorème 3.4.1 *Soit $\mu = (\Phi, \alpha, \beta) : \mathcal{I} \to \mathcal{I}'$ un morphisme d'institution tel qu'il existe un foncteur $\overline{\Phi}$ qui préserve les morphismes d'inclusion, que β_Σ soit surjectif pour toute signature $\Sigma \in |Sig|$, et que μ soit conservatif par rapport à $\overline{\Phi}$. Alors, si $\mathcal{I}$ possède la propriété de Beth, $\mathcal{I}'$ la possède également.*

Preuve Soit $\Sigma'_2 \setminus \Sigma'_1$ une signature implicitement définissable pour une Σ'_2-théorie T'. Cela signifie que pour tout Σ'_1-modèle $\mathcal{M}' \in |\mathcal{Mod}'(\Sigma'_1)|$ tel que pour tout modèles $\mathcal{M}'_1, \mathcal{M}'_2 \in \mathcal{Mod}'(T')$, tels que $\mathcal{Mod}'(\Sigma'_1 \hookrightarrow \Sigma'_2)(\mathcal{M}'_1) = \mathcal{Mod}'(\Sigma'_1 \hookrightarrow \Sigma'_2)(\mathcal{M}'_2) = \mathcal{M}'$, on a $\mathcal{M}'_1 \equiv_{\Sigma'_1} \mathcal{M}'_2$. Par le point 1. de la définition 3.4.1 il existe $\mathcal{M}_1 \in \beta^{-1}_{\overline{\Phi}(\Sigma'_2)}(\mathcal{M}'_1)$ et $\mathcal{M}_2 \in \beta^{-1}_{\overline{\Phi}(\Sigma'_2)}(\mathcal{M}'_2)$ tels que $\mathcal{Mod}(\overline{\Phi}(\sigma'))(\mathcal{M}_1) = \mathcal{Mod}(\overline{\Phi}(\sigma'))(\mathcal{M}_2)$. Soit alors $\varphi \in \mathcal{Sen}(\overline{\Phi}(\Sigma')) \setminus \alpha_{\overline{\Phi}(\Sigma')}(\mathcal{Sen}'(\Sigma'))$. Par le point 2. de la définition 3.4.1, il existe $E' \subseteq \mathcal{Sen}'(\Sigma')$ tel que $\mathcal{Mod}(\varphi) = \mathcal{Mod}(\alpha_{\overline{\Phi}(\Sigma')}(E'))$. Par conséquent, nous avons :

$$\begin{aligned}
\mathcal{M}_1 \models_{\overline{\Phi}(\Sigma'_2)} \varphi \quad & \Longleftrightarrow \mathcal{M}_1 \models_{\overline{\Phi}(\Sigma'_2)} \alpha_{\overline{\Phi}(\Sigma'_2)}(E') \\
& \Longleftrightarrow \mathcal{M}'_1 \models'_{\Sigma'_2} E' \text{ (condition de satisfaction)} \\
& \Longleftrightarrow \mathcal{M}'_2 \models'_{\Sigma'_2} E' \; (\mathcal{M}'_1 \equiv_{\Sigma'_2} \mathcal{M}'_2) \\
& \Longleftrightarrow \mathcal{M}_2 \models_{\overline{\Phi}(\Sigma'_2)} \alpha_{\overline{\Phi}(\Sigma'_2)}(E') \\
& \qquad\qquad\qquad \text{(condition de satisfaction)} \\
& \mathcal{M}_2 \models_{\overline{\Phi}(\Sigma'_2)} \varphi
\end{aligned}$$

Alors, $\overline{\Phi}(\Sigma'_2) \setminus \overline{\Phi}(\Sigma'_1)$ est explicitement définissable par rapport à $\alpha_{\overline{\Phi}(\Sigma'_2)}(T')$. De ce fait, pour toute formule $\varphi \in \mathcal{S}en'(\Sigma'_2 \setminus \Sigma'_1)$ il existe $E \subseteq \mathcal{S}en(\overline{\Phi}(\Sigma'_1))$ tel que $\mathcal{M}od(\alpha_{\overline{\Phi}(\Sigma'_2)}(T' \cup \{\varphi\})) = \mathcal{M}od(\alpha_{\overline{\Phi}(\Sigma'_i)}(T') \cup \mathcal{M}od(\mathcal{S}en(\overline{\Phi}(\Sigma'_1) \hookrightarrow \overline{\Phi}(\Sigma'_2))(E'))$. Par le point 2. de la définition 3.4.1, il existe un ensemble $E' \subseteq \mathcal{S}en'(\Sigma'_1)$ tel que $\mathcal{M}od(E) = \mathcal{M}od(\alpha_{\overline{\Phi}(\Sigma'_1)}(E'))$. Par la propriété de satisafaction des morphismes d'institutions, on a $\mathcal{M}od'(T' \cup \{\varphi\}) = \mathcal{M}od'(T' \cup E')$. □

3.5 Formulation institution-indépendante généralisée

Comme on peut le constater, les deux notions de définissabilité sont restreintes aux inclusions de signatures. Or, on a déjà vu qu'en théorie de la spécification il était intéressant de considérer d'autres relations entre les signatures que de simples inclusions. Cependant, il n'existe pas à notre connaissance de travaux traitant de ce problème. Nous généralisons donc les notions de définissabilité implicite et explicite à tout morphisme de signatures.

Définition 3.5.1 (définissabilité implicite) *Soit $\sigma : \Sigma \to \Sigma'$ un morphisme de signatures et soit T une Σ'-théorie. σ est dit* implicitement definissable par T *si pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ et pour tous modèles $\mathcal{M}', \mathcal{M}'' \in \mathcal{M}od(T)$ tels que $\mathcal{M}od(\sigma)(\mathcal{M}') = \mathcal{M}od(\sigma)(\mathcal{M}'') = \mathcal{M}$, on a $\mathcal{M}' \equiv_{\Sigma'} \mathcal{M}''$.*

Définition 3.5.2 (définissabilité explicite) *Soit $\sigma : \Sigma \to \Sigma'$ un morphisme de signature et soit T une Σ'-théorie. σ est dit* explicitement definissable par rapport à T *si pour toute formule $\varphi \in \mathcal{S}en(\Sigma') \setminus \mathcal{S}en(\sigma)(\Sigma)$ il existe $E \subseteq \mathcal{S}en(\Sigma)$ tel que les deux conditions suivantes soient vérifiées :*

1. *$T \cup \{\varphi\} \models_{\Sigma'} \mathcal{S}en(\sigma)(E)$;*
2. *$T \cup \mathcal{S}en(\sigma)(E) \models_{\Sigma'} \varphi$.*

Théorème 3.5.1 *Soit $\mathcal{I}$ une institution possédant l'interpolation de Craig sur une classe C de diagrammes. Soit $\sigma : \Sigma \to \Sigma'$ un morphisme de signature et soit T une Σ'-théorie. Alors, σ est implicitement définissable par rapport à T si et seulement si σ est explicitement définissable par rapport à T.*

Preuve La preuve est similaire à celle de la proposition 3.2.1 et celle du théorème 3.3.1. □

Deuxième partie

Lemme de consistance de Robinson et méthode des diagrammes

Ainsi que nous l'indiquions en introduction à la partie I, la théorie des institutions a été introduite dans le but de généraliser des résultats aussi bien en théorie de la spécification qu'en théorie des modèles, indépendamment des détails de la logique sous-jacente. Mais, alors que le premier de ces objectifs a inspiré de nombreux travaux, le deuxième ne fit l'objet que de quelques rares tentatives (*cf.* [130, 129, 131, 114, 113]). Or, l'intérêt de l'étude de la théorie des modèles dans le cadre institutionnel est multiple (*cf.* [43]) :

- elle fournit des résultats de théorie des modèles de façon générique ;
- elle exporte les méthodes modèles-théoriques des logiques classiques vers les autres logiques ;
- elle permet de faire de la théorie des modèles indépendamment du système logique sous-jacent ;

Alors que les deux premiers points ci-dessus sont d'un intérêt pragmatique, le dernier est plutôt d'ordre théorique. En effet, une telle approche de la théorie des modèles permet de s'abstraire des détails des logiques (en particulier de ceux de la logique du premier ordre classique) et de capturer l'essence même des concepts manipulés (*cf.* les définitions de définissabilité implicite et explicite [*cf.* défintions 3.5.1 et 3.5.2]). On parle alors de théorie abstraite des modèles par opposition avec la théorie standard des modèles consacrée à l'étude particulière des modèles de la logique du premier ordre classique. Bien sur, d'autres cadres que les institutions permettent de faire de la théorie abstraite des modèles. C'est le cas des logiques générales de Barwise par exemple (*cf.* [12]). Cependant, l'une des raisons faisant que le cadre des institutions est particulièrement bien adapté à l'étude de la théorie abstraite des modèles (ou théorie des modèles institution-indépendante) est qu'elle fournit la définition la plus complète (à notre connaissance) de cette théorie, incluant les notions de morphisme de signatures, de réduction de modèles et même de morphisme de systèmes logiques comme concepts premiers.

Nous avons vu dans la première partie que lorsqu'on dispose de la négation, la propriété d'interpolation de Craig est équivalente au lemme de consistence de Robinson (*cf.* théorème 2.2.2) et, par la proposition 1.4.2, que ce lemme découle presque directement de la propriété d'amalgation élémentaire (également appelée propriété de Robinson). Mais cette dernière propriété repose sur la notion d'équivalence élémentaire de modèles qui est souvent très difficile à démontrer dans la pratique. L'équivalence entre l'interpolation de Craig et le lemme de consistence de Robinson n'est donc que de peu d'utilité sans une méthode permettant de montrer que deux modèles sont élémentairement équivalents. La méthode des diagrammes, définie initialement dans le cadre de la théorie standard des modèles, est une méthode permettant justement de construire des modèles elémentairement équivalents à un modèle donné. C'est d'ailleurs à l'aide de cette méthode que Robinson a donné la preuve de son lemme de consistance dans le cadre de la logique du premier ordre classique. Puisqu'un des objectif de cette thèse est de fournir des conditions suffisantes pour montrer l'interpolation de Craig dans le cadre général des institutions, il est alors naturel de s'intéresser au lemme de consistence de Robinson et, en particulier, à la généralisation de la méthode des diagrammes dans ce cadre. Le problème qui se pose alors est que le cadre des institutions est trop général pour y définir la méthode des diagrammes. En effet, cette méthode repose sur

la notion de formules libres. Or, la définition institutionnelle de la satisfaction des formules par les modèles ne permet pas de considérer ce type de formules. Mais, si on s'intéresse à la logique du premier ordre, on s'aperçoit que la satisfaction des formules dans les structures du premier ordre est « stratifiée » par la notion d'interprétation des variables (dans le cas d'une logique modale, ce sont les états qui stratifient la satisfaction des formules). Or, la satisfaction des formules libres dépend justement de la notion d'interprétation des variables. Il est donc nécessaire de préciser la définition de la satisfaction des formules en la stratifiant par les objets permettant cette stratification. Cette notion de stratification interne d'une institution, formalisée par l'introduction du cadre des institutions stratifiées (*cf.* chapitre 1), nous permet alors de généraliser de façon institution-indépendante la notion d'extension élémentaire qui est à la base de la méthode des diagrammes (*cf.* chapitre 2). Nous généralisons alors la méthode des diagrammes en associant la notion de diagramme complet à celle d'extension élémentaire. Bien entendu, il n'est pas possible de définir des diagrammes complet dans le cadre de la logique modale. Cependant, notre généralisation de la notion d'extension élémentaire est indépendante de celle de diagramme et est donc valable pour cette dernière logique.
L'une des conditions suffisantes pour obtenir le lemme de consistance de Robinson est le théorème des chaînes élémentaires de Tarski. Dans [66], D. Găină et A. Popescu proposent une généralisation de la notion d'extension élémentaire s'appuyant sur les diagrammes définis par R. Diaconescu dans [40]. Ceci leur permet alors de donner une preuve institution-indépendante du théorème de Tarski ainsi que du lemme de consistance de Robinson. Cependant, leur travail concerne une classe, certes importante mais restreinte, de logiques puisqu'il suppose une structure interne du premier ordre très forte. Nous inspirant de ces travaux et nous basant sur notre propre généralisation de la notion d'extension élémentaire, nous étendons le cadre des institutions stratifiées de façon à prendre en compte la notion de constructeur abstrait de formules (*cf.* chapitre 3). Ceci nous permet alors de proposer une preuve du théorème de Tarski et du lemme de Robinson plus générales que celle de D. Găină et A. Popescu. En particulier, notre preuve du théorème de Tarski est valable dans le cadre des logiques modales.

Le plan de cette deuxième partie est le suivant. Dans le premier chapitre, nous introduisons la notion de stratification interne d'une institution et nous généralisons ensuite la méthode en introduisant le cadre des institutions stratifiées. Après un état de l'art sur la méthode des diagrammes où nous présentons les travaux de R. Diaconescu (*cf.* [40]) et de D. Găină et A. Popescu (*cf.* [66]), nous définissons dans le second chapitre les notions de diagramme complet et d'extension élémentaire. Nous appliquons ensuite cette généralisation de la méthode des diagrammes à deux résultats bien connus de théorie standard des modèles : les théorèmes de Löwenheim et Skolem et le test de Lòs-Vaught. Enfin, le dernier chapitre est consacré aux preuves institution-indépendantes du théorème des chaînes élémentaires de Tarski et du lemme de consistance de Robinson. Le cadre théorique choisi pour ces deux généralisations est une spécialisation des institutions stratifiées appelée κ-institution stratifiée permettant de prendre en compte la notion de constructeurs de formules dans les institutions stratifiées via une notion de foncteur sur les signatures.

Chapitre 1

Institutions stratifiées

1.1 Stratification interne

Nous avons vu en partie I, section 1.2 qu'il était possible de donner une définition institution-indépendante des connecteurs logiques ainsi que de la quantification via la notion de valuation abstraite des variables. Nous allons maintenant montrer qu'il est également possible de donner une définition institution-indépendante de la notion de satisfaction des formules via la notion d'état. Intuitivement, la notion de satisfaction des formules par les modèles d'une signature donnée repose sur la notion de *satisfaction interne* des formules « ouvertes » (l'équivalent des formules avec variables libres dans le cadre de la logique du premier ordre) paramétrée par les valuations abstraites des variables. La satisfaction des formules est alors *stratifiée* par cette notion de satisfaction interne et on parle de *stratification interne* d'une institution. Cette stratification interne se définit formellement de la façon suivante.

Définition 1.1.1 (stratification interne) *Soit $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Une* stratification interne de $\mathcal{I}$ *est la donnée d'un quadruplet $\mathcal{St}(\mathcal{I}) = (\mathcal{Sig}_{\mathcal{St}}, \mathcal{Sen}_{\mathcal{St}}, \mathcal{Mod}_{\mathcal{St}}, [\![]\!]_{\mathcal{St}})$ tel que :*

- *$\mathcal{Sig}_{\mathcal{St}}$ est la* catégorie *dont :*
  - *les objets sont les morphismes de signatures $\chi : \Sigma \to \Sigma'$ de $\mathcal{I}$,*
  - *les morphismes $\sigma : (\chi_1 : \Sigma_1 \to \Sigma'_1) \to (\chi_2 : \Sigma_2 \to \Sigma'_2)$ sont des couples $(\sigma_1 : \Sigma_1 \to \Sigma_2, \sigma'_1 : \Sigma'_1 \to \Sigma'_2)$ de morphismes de signatures de $\mathcal{I}$.*
- *$\mathcal{Sen}_{\mathcal{St}} : \mathcal{Sig}_{\mathcal{St}} \to \mathcal{Set}$ est un* foncteur *qui à toute signature $\chi : \Sigma \to \Sigma' \in |\mathcal{Sig}_{\mathcal{St}}|$ associe l'ensemble $\mathcal{Sen}(\Sigma') \in |\mathcal{Set}|$ des Σ'-formule dans $\mathcal{I}$;*
- *$\mathcal{Mod}_{\mathcal{St}} : \mathcal{Sig}_{\mathcal{St}} \to \mathcal{Cat}^{op}$ est un* foncteur *qui à toute signature $\chi : \Sigma \to \Sigma' \in |\mathcal{Sig}_{\mathcal{St}}|$ associe la catégorie $\mathcal{Mod}(\Sigma) \in |\mathcal{Cat}|$ des Σ-modèles dans $\mathcal{I}$;*
- *$[\![]\!]_{\mathcal{St}} = ([\![]\!]^{\mathcal{St}}_{\chi})_{\chi \in |\mathcal{Sig}_{\mathcal{St}}|}$ est une* famille de foncteurs *telle que pour toute signature $\chi \in |\mathcal{Sig}_{\mathcal{St}}|$, le foncteur $[\![]\!]^{\mathcal{St}}_{\chi} : \mathcal{Mod}_{\mathcal{St}}(\chi) \to \mathcal{Set}$ associe à tout modèle $\mathcal{M} \in |\mathcal{Mod}_{\mathcal{St}}(\chi)|$ l'ensemble $[\![\mathcal{M}]\!]^{\mathcal{St}}_{\chi} = \{\mathcal{M}' / \mathcal{Mod}(\chi)(\mathcal{M}') = \mathcal{M}\}$.*

La stratification interne d'une institution permet de préciser la définition de la relation de satisfaction des formules par les modèles de la façon suivante :

soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution et $\mathcal{S}t(\mathcal{I}) = (\mathcal{S}ig_{\mathcal{S}t}, \mathcal{S}en_{\mathcal{S}t}, \mathcal{M}od_{\mathcal{S}t}, [\![]\!]_{\mathcal{S}t})$ sa stratification interne. Alors, pour toute signature $(\chi : \Sigma \to \Sigma') \in |\mathcal{S}ig_{\mathcal{S}t}|$, tout χ-modèle $\mathcal{M} \in |\mathcal{M}od_{\mathcal{S}t}(\chi)|$ et toute χ-formule $\varphi' \in \mathcal{S}en_{\mathcal{S}t}(\chi)$, on a :

$$\mathcal{M} \models_\Sigma \varphi' \Leftrightarrow \forall \mathcal{M}' \in [\![\mathcal{M}]\!]^{\mathcal{S}t}_\chi,\ \mathcal{M}' \models_{\Sigma'} \varphi'$$

Intuitivement, la définition précédente indique que la notion de variable d'une institution donnée peut être représentée par une classe particulière de morphismes de signatures (les objets de la catégorie $\mathcal{S}ig_{\mathcal{S}t}$). Dans le cas de la logique du premier ordre par exemple, les variables sont en fait des symboles de fonction d'arité 0. Elles sont alors naturellement représentées par la classe des morphismes de signatures réalisant l'identité sur les symboles de sortes et de prédicats et une inclusion sur les symboles de fonctions déterminée par l'union de l'ensemble initial avec un ensemble de symboles de fonction d'arité 0 (ce sont les morphismes représentables introduits à la définition 1.2.7, partie I). Dans le cas de la logique du second ordre cette inclusion est déterminée par l'union de l'ensemble initial de symboles de fonction avec un autre ensemble de symboles de fonctions de toute arité.

Exemple 1.1.1 *Soit FOL l'institution de la logique des prédicats du premier ordre définie dans l'exemple 1.1.3, partie I. Si on se restreint, pour les variables internes, au symboles de constantes, alors $\mathcal{S}t(FOL)$ est la stratification usuelle de FOL, où les signatures sont des morphismes de signatures $\sigma : (\mathcal{S}, \mathcal{F}, \mathcal{P}) \to (\mathcal{S}, \mathcal{F} \cup \mathcal{X}, \mathcal{P})$ ($\mathcal{X}$ est l'ensemble de variables associé à $(\mathcal{S}, \mathcal{F}, \mathcal{P})$) ; les $(\mathcal{S}, \mathcal{F}, \mathcal{P})$-formules sont les $(\mathcal{S}, \mathcal{F} \cup \mathcal{X}, \mathcal{P})$-formules. Un état associé à un $(\mathcal{S}, \mathcal{F}, \mathcal{P})$-modèle $\mathcal{M}$ est alors la donnée d'une valuation $\nu : X \to M$ des variables, i.e. $[\![\mathcal{M}]\!] = M^X$. En effet, une valuation des variables est simplement la donnée d'une valeur $m \in M$ à tout symbole $x \in X$. La donnée d'une $(\mathcal{S}, \mathcal{F} \cup \mathcal{X}, \mathcal{P})$-extension de $\mathcal{M}$ est alors bien équivalente à la donnée d'une application de X dans M.*

1.2 Définition et exemples

Le concept de stratification peut être défini de façon plus abstraite que par la notion de stratification interne d'institution. Cette dernière notion présente en effet l'inconvénient de requérir que les éléments utilisés pour « stratifier » la satisfaction des formules possèdent un équivalent syntaxique. Or, dans le cas d'une logique modale propositionnelle, la notion d'état ne possède pas d'équivalent syntaxique. Il n'est donc pas possible de la capturer par une classe particulière de morphismes et la stratification interne d'une telle logique selon la définition 1.1.1 fait qu'à toute structure de Kripke $\mathcal{M} = (\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$, on a $[\![\mathcal{M}]\!] = \{\mathcal{M}\}$. Intuitivement, nous voudrions $[\![\mathcal{M}]\!] = \mathfrak{E}$ puisque ce sont les états qui stratifient la satisfaction des formules. Afin de remédier à ce problème, nous introduisons alors le cadre des institutions stratifiées. En plus de définir les formules et modèles associés à une signature donnée, ce cadre va nous permettre de définir de façon générale la stratification de la relation de satisfaction, *i.e.* indépendamment de la nature des objets utilisés pour la validation des formules.

Définition 1.2.1 (pré-institution stratifiée) *Une* pré-institution stratifiée $\mathcal{SI}$ *est la donnée d'un quadruplet* $(\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ *tel que :*

- *le triplet* $(\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od)$ *est défini comme dans le cas institutionel (cf. définition 1.1.1, partie I) ;*
- $[\![]\!] = ([\![]\!]_\Sigma)_{\Sigma \in |\mathcal{S}ig|}$ *est une* famille de foncteurs *indéxée par* $\mathcal{S}ig$ *telle que pour toute signature* $\Sigma \in |\mathcal{S}ig|$, $[\![]\!]_\Sigma : \mathcal{M}od(\Sigma) \to \mathit{Set}$ *est associe à tout* Σ*-modèle* $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ *l'ensemble des* états *associés à* $\mathcal{M}$.

Pour toute signature $\Sigma \in |\mathcal{S}ig|$, on définit alors la relation de satisfaction $\models_\Sigma \subseteq \mathcal{M}od(\Sigma) \times \mathcal{S}en(\Sigma)$ de la façon suivante :

$$\mathcal{M} \models_\Sigma \varphi \Longleftrightarrow \forall \eta \in [\![\mathcal{M}]\!]_\Sigma,\ \mathcal{M} \models_{\Sigma,\eta} \varphi$$

où $\mathcal{M} \models_{\Sigma,\eta} \_ \subseteq \mathcal{S}en(\Sigma)$ est une relation unaire définie sur $\mathcal{S}en(\Sigma)$.

Exemple 1.2.1 (logique propositionnelle) *La pré-institution stratifiée de la logique propositionnelle est définie comme suit :*

- $\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od$ *sont définis comme dans l'exemple 1.1.1, partie I;*
- *étant donnés une signature* Σ *et un* Σ*-modèle* $v : \Sigma \to \{0,1\}$, $[\![v]\!]_\Sigma = \mathbb{1}$ *où* $\mathbb{1}$ *est l'objet terminal de* $\mathcal{S}et$*. Aucun objet ne stratifie donc la validation des formules et la relation* $v \models_{\Sigma,\mathbb{1}}$ *est définie inductivement sur la structure des formules propositionnelles de la façon suivante :*
  - $\forall p \in \Sigma,\ v \models_{\Sigma,\mathbb{1}} p$ *ssi* $v(p) = 1$,
  - $\forall \varphi \in \mathcal{S}en(\Sigma),\ v \models_{\Sigma,\mathbb{1}} \neg\varphi$ *ssi* $v \not\models_{\Sigma,\mathbb{1}} \varphi$,
  - $\forall \varphi, \psi \in \mathcal{S}en(\Sigma),\ v \models_{\Sigma,\mathbb{1}} \varphi \vee \psi$ *ssi* $v \models_{\Sigma,\mathbb{1}} \varphi$ *ou* $v \models_{\Sigma,\mathbb{1}} \psi$.

Exemple 1.2.2 (logique propositionnelle modale) *La pré-institution stratifiée de la logique propositionnelle modale est définie comme suit :*

- $\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od$ *sont définis comme dans l'exemple 1.1.2, partie I;*
- *étant donnés une signature* Σ *et un* Σ*-modèle* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$, $[\![(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})]\!]_\Sigma = \mathfrak{E}$ *et, pour tout* $\eta \in \mathfrak{E}$, $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta}$ *est définie inductivement sur la structure des formules de la façon suivante :*
  - $\forall p \in \Sigma,\ (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} p$ *ssi* $\mathfrak{v}(\eta, p) = 1$,
  - $\forall \varphi \in \mathcal{S}en(\Sigma),\ (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \neg\varphi$ *ssi* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \not\models_{\Sigma,\eta} \varphi$,
  - $\forall \varphi \in \mathcal{S}en(\Sigma),\ (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \Box\varphi$ *ssi pour tout* $\eta' \in \mathfrak{E}$ *tel que* $\eta \mathfrak{R} \eta'$, *on a* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta'} \varphi$,
  - $\forall \varphi, \psi \in \mathcal{S}en(\Sigma),\ (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \varphi \vee \psi$ *ssi* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \varphi$ *ou* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \psi$.

  On a alors bien $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models^{\mathrm{PML}}_\Sigma \varphi$ *ssi pour tout* $\eta \in \mathfrak{E}$ *on a* $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma,\eta} \varphi$.

Exemple 1.2.3 (logique du premier ordre) *La pré-institution stratifiée de la logique du premier ordre est définie comme suit :*

- $\mathcal{S}ig$ *et* $\mathcal{M}od$ *sont définis comme indiqué dans l'exemple 1.1.3, partie I;*
- $\mathcal{S}en : \mathcal{S}ig \to \mathcal{S}et$ *est un foncteur qui associe à toute signature* $\Sigma \in |\mathcal{S}ig|$ *l'ensemble de toutes les formules construites sur* Σ *(i.e. y compris les formules libres) ;*

- *étant donnés une signature* $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$*, un ensemble de variables* $\mathcal{X}$ *tel que* $\mathcal{X} \cap (\mathcal{S} \cup \mathcal{F} \cup \mathcal{R}) = \varnothing$ *et une* Σ*-structure* $\mathcal{M}$*,* $[\![\mathcal{M}]\!]_\Sigma = M^{\mathcal{X}}$ *et, pour toute interprétation de variables* $\nu \in M^{\mathcal{X}}$*,* $\mathcal{M} \models_{\Sigma,\nu}$ *est définie inductivement sur la structure des formules de la façon suivante :*
  - $\forall t, t' \in T_\Sigma(\mathcal{X})$, $\mathcal{M} \models_{\Sigma,\nu} t = t'$ *ssi* $\nu(t) = \nu(t')$,
  - $\forall r \in \mathcal{R}, \forall t_1, \ldots, t_n \in T_\Sigma(\mathcal{X})$, $\mathcal{M} \models_{\Sigma,\nu} r(t_1, \ldots, t_n)$ *ssi*
  $$(\nu(t_1), \ldots, \nu(t_n)) \in r^{\mathcal{M}}$$
  - $\forall \varphi \in \mathcal{Sen}(\Sigma)$, $\mathcal{M} \models_{\Sigma,\nu} \neg\varphi$ *ssi* $\mathcal{M} \not\models_{\Sigma,\nu} \varphi$,
  - $\forall x \in \mathcal{X}, \forall \varphi \in \mathcal{Sen}(\Sigma)$, $\mathcal{M} \models_{\Sigma,\nu} (\forall x)\varphi$ *ssi* $\mathcal{M} \models_{\Sigma,\nu'} \varphi$ *pour toute interprétation* ν' *tel que* $\nu'(y) = \nu(y)$ *pour toute variable* $y \in \mathcal{X}$*, sauf éventuellement pour* x*,*
  - $\forall \varphi, \psi \in \mathcal{Sen}(\Sigma)$, $\mathcal{M} \models_{\Sigma,\nu} \varphi \vee \psi$ *ssi* $\mathcal{M} \models_{\Sigma,\nu} \varphi$ *ou* $\mathcal{M} \models_{\Sigma,\nu} \psi$

Exemple 1.2.4 (logique observationnelle [68]) *La pré-institution stratifiée de la logique observationnelle est définie de la façon suivante :*

- *Les signatures sont des triplets* $(\mathcal{S}, \mathcal{F}, \mathcal{X})$ *où* $\mathcal{S}$ *est un ensemble de sortes parmi lesquelles on distingue un sous-ensemble* S_{obs} *de sortes observables ;* $\mathcal{F}$ *est un ensemble de noms de fonctions, tous équipés d'un profil sur* $\mathcal{S}^* \times \mathcal{S}$ *;* $\mathcal{X}$ *est un ensemble indéxé par* $\mathcal{S}$ *de variables. Dans la suite, nous noterons* $f : s_1 \times \ldots \times s_n \to s$ *toute fonction* $f \in \mathcal{F}$ *dont le profil est* $(s_1, \ldots, s_n, s)$*. Les morphismes de signatures sont toutes les applications respectant les ensembles et préservant les sortes observables et les profils ;*
- $\mathcal{Sen} : \mathcal{Sig} \to \mathcal{Set}$ *est défini de la même façon que dans l'exemple 1.2.3 mis à part le fait qu'on restreint les atomes aux équations, i.e. aux paires* (t, t') *de termes de même sorte. Dans la suite, de telles paires de termes seront notées* $t = t'$ *;*
- $\mathcal{Mod} : \mathcal{Sig} \to \mathcal{Cat}^{op}$ *est le foncteur qui associe :*
  - *à toute signature* $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{X})$ *la catégorie des* Σ*-algèbres multi-sortes,*
  - *à tout morphisme de signatures* $\sigma : \Sigma \to \Sigma'$ *le foncteur d'oubli* $\mathcal{Mod}(\sigma) : \mathcal{Mod}(\Sigma') \to \mathcal{Mod}(\Sigma)$ *qui associe à toute* Σ'*-algèbre* $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ *la* Σ*-algèbre* $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$ *telle que pour tout* $s \in \mathcal{S}$*,* $M_s = M'_{\sigma(s)}$ *et, pour tout symbole de fonction* $f : s_1 \times \ldots \times s_n \to s \in F$*,* $f^{\mathcal{M}} = \sigma(f)^{\mathcal{M}'}$*.*

Étant donnée une signature $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{X})$*, on définit la signature* $\Sigma_\Box = (\mathcal{S}, \mathcal{F} \cup \{\Box_s \mid s \in \mathcal{S} \setminus \mathcal{S}_{obs}\}, \mathcal{X})$*. Un* Σ-contexte C *est un terme de* $T_{\Sigma_\Box}(\mathcal{X})_s$ $(s \in \mathcal{S}_{obs})$ *qui est l'ensemble des termes construits sur la signature* $\Sigma_\Box$ *de sorte* s *ne contenant qu'une seule occurence d'un symbole* $\Box_{s'}$*, celui-ci étant l'unique symbole de l'ensemble* $\Box = \{\Box_s \mid s \in S \setminus S_{obs}\}$ *apparaissant dans* C*. Si l'unique symbole de* $\Box$ *apparaissant dans un contexte* C *est* $\Box_s$*, alors* C *est de sorte* s*, noté* $C : s$*. L'application d'un contexte* $C : s$ *à un terme* $t \in T_\Sigma(\mathcal{X})_s$*, notée* $C[t]$*, est le terme obtenu en substitutant le terme* t *pour* $\Box_s$*. On note* $\mathcal{Ctx}$ *l'ensemble des contextes* $C : s$*. Étant donnée une* Σ*-algèbre* $\mathcal{A}$*,* $[\![\mathcal{A}]\!] = M^{\mathcal{X}} \times \mathcal{Ctx}$*.*

$\mathcal{A} \models_{\Sigma,(\nu,C)} t = t'$ *(*$C : s'$*) si et seulement si :*

$$\begin{cases} \nu^\#(t) = \nu^\#(t') \text{ si } t, t' \in T_\Sigma(\mathcal{X})_s \text{ et } s \in S_{obs} \\ \nu^\#(C[t]) = \nu^\#(C[t']) \text{ si } t, t' \in T_\Sigma(\mathcal{X})_{s'} \\ true \text{ sinon} \end{cases}$$

Les connecteurs et les quantificateurs sont évalués de façon habituelle.

Exemple 1.2.5 (logique du second ordre) *La pré-institution stratifiée de la logique du second ordre est définie de la même façon que la pré-institution stratifiée de la logique du premier ordre à ceci près que les variables sont désormais mises pour des symboles de fonction de toute arité.*

Proposition 1.2.1 *Soient $\mathcal{I} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ une institution. Sa stratification interne $\mathcal{St}(\mathcal{I})$ est une pré-institution stratifiée.*

Preuve Découle directement des définitions. □

Définition 1.2.2 (institution stratifiée) *Une* institution stratifiée $\mathcal{SI}$ *est une pré-institution stratifiée vérifiant la condition de satisfaction, i.e. :*
$\forall \sigma \in \mathcal{Sig}(\Sigma, \Sigma'), \forall \mathcal{M}' \in |\mathcal{Mod}(\Sigma')|, \forall \varphi \in \mathcal{Sen}(\Sigma)$,

$$\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma)(\varphi) \Leftrightarrow \mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma} \varphi$$

Remarque 1.2.1 *Il ne faut pas confondre les institutions stratifiées avec les institutions concrètes (cf. [17]). En effet, ces dernières enrichissent les institutions avec la notion d'ensemble de base pour les modèles alors que les institutions stratifiées enrichissent ces dernières avec la notion d'état (ou de valuation des variables).*

Les exemples de pré-institutions stratifiées donnés plus haut sont tous des institutions stratifiées à l'exception de la logique observationnelle. En effet, on a le contre-exemple suivant.

Contre-exemple 1.2.1 *Soit $\Sigma_1 = (\mathcal{S}_1, \mathcal{F}_1, \mathcal{X}_1)$ la signature observationnelle telle que $\mathcal{S}_1 = \{\mathtt{thesort}, \mathtt{bool}\}$, $\mathcal{S}_{1_{obs}} = \{\mathtt{bool}\}$, $\mathcal{F}_1 = \{\mathtt{thelabel} : \mathtt{thesort} \to \mathtt{bool}\}$ et $\mathcal{X}_{1_{\mathtt{thesort}}} = \{x, y\}$.*
Soit $\Sigma_2 = (\mathcal{S}_2, \mathcal{F}_2, \mathcal{X}_2)$ la signature observationnelle telle que $\mathcal{S}_2 = \mathcal{S}_1$, $\mathcal{S}_{2_{obs}} = \mathcal{S}_{1_{obs}}$, $\mathcal{F}_2 = \mathcal{F}_1 \cup \{\mathtt{thelabel}' : \mathtt{thesort} \to \mathtt{bool}\}$ et $\mathcal{X}_{2_{\mathtt{thesort}}} = \mathcal{X}_{1_{\mathtt{thesort}}}$.
Il est clair que Σ_1 est incluse dans Σ_2. Soit maintenant $\mathcal{A}$ la Σ_2-algèbre définie par $A_{\mathtt{thesort}} = \{a, b\}$, $\mathtt{thelabel}^{\mathcal{A}} = \{a, b\}$ et $\mathtt{thelabel}'^{\mathcal{A}} = \{a\}$ (on rappelle que $\mathcal{C}tx = \{\mathtt{thelabel}(\square_{\mathtt{thesort}}), \mathtt{thelabel}'(\square_{\mathtt{thesort}})\}$). La Σ_1-algèbre $\mathcal{B} = \mathcal{Mod}(\Sigma_1 \subseteq \Sigma_2)(\mathcal{A})$ est caractérisée par

$$B_{\mathtt{thesort}} = \{a, b\},\ \mathtt{thelabel}^{\mathcal{B}} = \{a, b\}$$

La formule $x = y$ est donc satisfaite dans $\mathcal{B}$ mais pas dans $\mathcal{A}$ car $b \notin \mathtt{thelabel}'^{\mathcal{A}}$ ce qui remet en cause la condition de satisfaction

Sous une hypothèse très simple, les pré-institutions stratifiées présentent les institutions stratifiées comme l'indique la proposition suivante.

Proposition 1.2.2 *Toute pré-institution stratifiée $\mathcal{SI} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, [\![]\!])$ telle que pour tout morphisme de signatures $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$ et tout Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ il existe une transformation naturelle $[\![]\!]_\sigma : [\![]\!]_{\Sigma'} \Rightarrow [\![]\!]_{\Sigma} \circ \mathcal{Mod}(\sigma)$ satisfaisant les deux conditions suivantes :*

1. *pour tout Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$, $[\![\mathcal{M}']\!]_\sigma$ est une application surjective ;*
2. *pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ et tout $\eta' \in [\![\mathcal{M}']\!]_{\Sigma'}$:*

$$\mathcal{M}' \models_{\Sigma', \eta'} \mathcal{Sen}(\sigma)(\varphi) \Longleftrightarrow \mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma, [\![\mathcal{M}']\!]_\sigma(\eta')} \varphi$$

est une institution stratifiée.

Preuve Il suffit pour cela de montrer que la condition de satisfaction est vérifiée, *i.e.* :
$\forall \sigma \in \mathcal{Sig}(\Sigma, \Sigma'), \forall \mathcal{M}' \in |\mathcal{Mod}(\Sigma')|, \forall \varphi \in \mathcal{Sen}(\Sigma)$,

$$\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\varphi) \Leftrightarrow \mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma} \varphi$$

Soient donc $\mathcal{M}' \in |\mathcal{Mod}(\Sigma')|$ un Σ'-modèle et $\varphi \in \mathcal{Sen}(\Sigma)$ une Σ-formule.

($\Rightarrow$) Supposons $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma)(\varphi)$ et soit $\eta \in [\![\mathcal{Mod}(\sigma)(\mathcal{M}')]\!]_{\Sigma}$. Puisque $[\![\mathcal{M}']\!]_{\sigma}$ est surjective, il existe $\eta' \in [\![\mathcal{M}']\!]_{\Sigma'}$ tel que $[\![\mathcal{M}']\!]_{\sigma}(\eta') = \eta$. Par hypothèse, $\mathcal{M}' \models_{\Sigma',\eta'} \mathcal{Sen}(\sigma)(\varphi)$. Par la condition 2 ci-dessus, on a alors $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma,\eta} \varphi$.

($\Leftarrow$) Supposons $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma} \varphi$ et soit $\eta' \in [\![\mathcal{M}']\!]_{\Sigma'}$. Par hypothèse, on a $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma,[\![\mathcal{M}']\!]_{\sigma}(\eta')} \varphi$. Par le point 2 ci-dessus, on en déduit alors $\mathcal{M}' \models_{\Sigma',\eta'} \mathcal{Sen}(\sigma)(\varphi)$. Puisque $[\![\mathcal{M}']\!]_{\sigma}$ est une application (*i.e.* une fonction totale), on en conclut alors $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\varphi)$.

□

Exemple 1.2.6 *En logique propositionnelle, étant donnés un morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ et un Σ_2-modèle $\nu' \in |\mathcal{Mod}(\Sigma')|$, $[\![\nu']\!]_{\sigma} : [\![\nu']\!] \to [\![\mathcal{Mod}(\sigma)(\nu')]\!]$ est l'identité $Id_{\mathbb{I}}$. De plus, pour tout modèle, le comportement de chaque connecteur est défini par sa table de vérité. De ce fait, $\nu' \models \mathcal{Sen}(\varphi)$ est équivalent à $\mathcal{Mod}(\sigma)(\nu') \models \varphi$.*

Exemple 1.2.7 *En logique propositionnelle modale, étant donnés un morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ et un Σ_2-modèle $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$, $[\![(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})]\!]_{\sigma} : [\![(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})]\!] \to [\![(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}_{|\Sigma_1})]\!]$ est $Id_{\mathfrak{E}}$ (l'identité sur $\mathfrak{E}$). Puisque la relation d'accessiblité $\mathfrak{R}$ est préservée par le foncteur de réduction, le comportement des modalités est préservée. On a donc $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \models_{\Sigma_2,\eta} \mathcal{Sen}(\sigma)(\varphi)$ équivalent à $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v}_{|\Sigma_1}) \models_{\Sigma_1,\eta} \varphi$.*

Exemple 1.2.8 *En logique du premier ordre, étant donnés un morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ et un Σ_2-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma_2)|$, on définit la relation d'équivalence $\equiv$ sur $M'^{\mathcal{X}_2}$ de la façon suivante :*

$$\nu \equiv \nu' \Leftrightarrow (\forall x \in \mathcal{X}_1,\ \nu(\sigma(x)) = \nu'(\sigma(x)))$$

Il est évident que $M'^{\mathcal{X}_2}/_{\equiv}$ et $M'^{\mathcal{X}_1}$ sont isomorphes. Soit donc $[\![\mathcal{M}']\!]_{\sigma}$ l'application quotient associée à $\equiv$. Par définition $[\![\mathcal{M}']\!]_{\sigma}$ est surjective. On montre alors la propriété attendue par simple induction sur la structure des formules.

Remarque 1.2.2 *Dans l'exemple qui précède, il convient d'imposer soit un ensemble de variables commun à toutes les signatures, soit qu'il existe une injection de l'ensemble de variables associé à Σ_1 dans celui associé à Σ_2. Ceci est très important comme le montre le contre exemple suivant.*

Contre-exemple 1.2.2 *Soient $\Sigma_1 = (\emptyset, \{=^2\}, \{x, y\})$ et $\Sigma_2 = (\emptyset, \{=^2\}, \{x\})$ des signatures. Soit $\sigma : \Sigma_1 \to \Sigma_2$ le morphisme de signatures défini comme l'identité sur l'ensemble des prédicats et associant x à x et y. Soit $\mathcal{M}'$ le Σ_2-modèle dont l'ensemble de base est $\{a, b\}$ et tel que $=^{\mathcal{M}}= \{(a, a), (b, b)\}$. Soit $\varphi = \exists x, \exists y, \neg x = y$. Alors, $\mathcal{Mod}(\sigma)(\mathcal{M}') \models_{\Sigma_1} \varphi$ mais, $\mathcal{M}' \not\models_{\Sigma_2} \exists x, \exists x, \neg x = x$.*

Le contre-exemple suivant indique indique la raison pour laquelle la pré-institution stratifiée de la logique observationnelle ne présente pas une institution (*cf.* contre-exemple 1.2.1).

Exemple 1.2.9 *En logique observationnelle, étant donnés un morphisme de signatures $\sigma : \Sigma_1 \to \Sigma_2$ et un Σ_2-modèle $\mathcal{M}'$, aucune application $[\![\mathcal{M}']\!]_\sigma$ ne peut être définie car $\sigma^\sharp(\mathcal{C}tx) \subset \mathcal{C}tx'$. Le seul moyen de définir une telle application est d'imposer $\sigma^\sharp(\mathcal{C}tx) = \mathcal{C}tx'$* [1]*, i.e. pour toute $f : s_1 \times \ldots \times s_n \to s \in F_2$, où $s \in S_{2_{obs}}$, il existe $f' \in F_1$ telle que $\sigma(f') = f$. C'est de cette façon que les morphismes de signatures ont été définis dans [68] de façon à ce que la condition de satisfaction soit vérifiée.*

Tout l'intérêt des institutions stratifiées par rapport aux institutions introduites au chapitre 1, partie I est de pouvoir considérer la notion de formules ouvertes [2]. En effet, la définition de la satisfaction dans les institutions restreint de fait l'ensemble des formules d'une signature donnée à l'ensemble des formules closes construites sur cette signature. Ainsi, pour un modèle donné, une formule est ou bien vraie ou bien fausse. Dans le cadre de la logique du premier ordre par exemple, ceci signifie qu'une formule est vraie (resp. fausse) pour toute interprétation des variables dans ce modèle. Or, la présence de variables libres dans les formules ouvertes a pour conséquence que ces dernières peuvent être vraies pour certaines interprétation des variables et fausses pour d'autres. C'est le cas par exemple de la formule $(\forall x) x + y > x$ qui est vraie dans l'ensemble des entiers relatifs pour toute interprétation ν telle que $\nu(y) > 0$ et fausse sinon. Elle n'est donc ni vrai ni fausse dans $\mathbb{Z}$. Nous verrons dans la section suivante l'importance de cette notion de formules ouvertes pour la généralisation de la notion de diagramme complet.

La preuve d'équivalence entre l'interpolation de Craig et la propriété de Robinson que nous avons donnée en partie I, section 2.2.2 suppose que l'institution considérée soit fermée par négation. Dans le cadre des institutions stratifiées, cette notion de négation se définit naturellement de la façon suivante.

Définition 1.2.3 (négation (sémantique)) *Soit $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ une institution stratifiée et soit $\Sigma \in |\mathcal{S}ig|$ une signature. Une Σ-formule $\varphi' \in \mathcal{S}en(\Sigma)$ est la* négation (sémantique) *d'une formule $\varphi \in \mathcal{S}en(\Sigma)$ si la condition suivante est vérifiée :*

$$\forall \mathcal{M} \in |\mathcal{M}od(\Sigma)|, \forall \eta \in [\![\mathcal{M}]\!]_\Sigma,\ \mathcal{M} \models_{\Sigma,\eta} \varphi' \Leftrightarrow \mathcal{M} \not\models_{\Sigma,\eta} \varphi$$

L'institution stratifiée $\mathcal{SI}$ est dite fermée par négation (sémantique) *si pour toute signature $\Sigma \in |\mathcal{S}ig|$, toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$ possède une négation (sémantique).*

Il est clair que la négation est transportée le long des morphismes de signatures comme le montre la proposition suivante.

Proposition 1.2.3 *Soit $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ une institution stratifiée. Pour tout morphisme de signature $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$, tout Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, $\mathcal{M}' \models_{\Sigma'} \neg \mathcal{S}en(\sigma)(\varphi)$ alors $\mathcal{M}' \models_{\Sigma'} \mathcal{S}en(\sigma)(\neg\varphi)$.*

1. $\sigma^\sharp$ est l'extension canonique de σ aux termes dans $T_{\Sigma_\square}(\mathcal{X})$.

2. On rappelle que dans le cadre de la logique du premier ordre, les formules ouvertes sont toutes celles dont certaines occurences de variables ne sont pas dans la portée d'un quantificateur.

Preuve On a :

$$\begin{aligned}
\mathcal{M}' \models_{\Sigma'} \neg \mathcal{S}en(\sigma)(\varphi) & \Leftrightarrow \forall \eta' \in [\![\mathcal{M}]\!]_{\Sigma'},\ \mathcal{M}' \models_{\Sigma',\eta'} \neg \mathcal{S}en(\sigma)(\varphi) \\
& \Leftrightarrow \forall \eta' \in [\![\mathcal{M}]\!]_{\Sigma'},\ \mathcal{M}' \not\models_{\Sigma',\eta'} \mathcal{S}en(\sigma)(\varphi) \\
& \Leftrightarrow \forall \eta' \in [\![\mathcal{M}]\!]_{\Sigma'},\ \mathcal{M}od(\sigma)(\mathcal{M}') \not\models_{\Sigma,[\![\mathcal{M}]\!]_\sigma(\eta')} \varphi \\
& \Leftrightarrow \forall \eta' \in [\![\mathcal{M}]\!]_{\Sigma'},\ \mathcal{M}od(\sigma)(\mathcal{M}') \models_{\Sigma,[\![\mathcal{M}]\!]_\sigma(\eta')} \neg\varphi \\
& \Leftrightarrow \forall \eta' \in [\![\mathcal{M}]\!]_{\Sigma'},\ \mathcal{M}' \models_{\Sigma',\eta'} \mathcal{S}en(\sigma)(\neg\varphi) \\
& \Leftrightarrow \mathcal{M}' \models_{\Sigma'} \mathcal{S}en(\sigma)(\neg\varphi)
\end{aligned}$$

□

La définition de la négation dans les institutions stratifiées correspond bien à l'intuition qu'on en a dans le sens où une formule φ est la négation d'une formule ψ si elle est vraie partout où ψ est fausse et inversement. Seulement, lorsqu'on considère la stratification de la logique du premier ordre cette définition de la négation ne vérifie pas l'équation 1.1 ci-dessous qui est fondamentale pour montrer que dans toute institution possédant la négation, toute théorie complète posséde une classe de modèles deux à deux élémentairement équivalents.

$$\mathcal{M}od(\neg\varphi) = \mathcal{M}od(\Sigma) \setminus \mathcal{M}od(\varphi) \tag{1.1}$$

Or, c'est grâce à cette dernière propriété que nous pouvons montrer l'équivalence entre l'interpolation de Craig et la consistance de Robinson. En fait, cette équation n'est vérifiée dans la logique du premier ordre que pour les formules closes. De façon à retrouver la propriété que toute théorie complète possède, en présence de la négation, une classe de modèles deux à deux élémentairement équivalents, il est nécessaire de capturer le concept de formule close et de redéfinir les notions de théories et d'équivalence élémentaire de modèles en fonction.

Définition 1.2.4 (sémantiquement fermée) *Soit une institution stratifiée $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ et soit $\Sigma \in |\mathcal{S}ig|$ une signature. Une Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$ est dite* sémantiquement fermée *si pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$, l'ensemble $\{\eta \in [\![\mathcal{M}]\!] / \mathcal{M} \models_{\Sigma,\eta} \varphi\}$ est soit égal à $[\![\mathcal{M}]\!]_\Sigma$ soit vide.*
Dans la suite, on notera $\mathcal{S}en_{\mathcal{F}}(\Sigma)$ l'ensemble des Σ-formules sémantiquement fermées.

La notion de formules sémantiquement fermées caractérise en fait plus de formules que les seules formules closes. En effet, si on se place dans le cadre de la logique du premier ordre les formules sémantiquement fermées sont les formules closes plus toutes les tautologies (et leurs négations). Ceci est cependant suffisant pour notre propos comme le montre la proposition suivante.

Proposition 1.2.4 *Pour toute institution stratifiée $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ fermée par négation l'équivalence suivante est vérifiée :*

$$(\mathcal{M}od(\neg\varphi) = \mathcal{M}od(\Sigma) \setminus \mathcal{M}od(\varphi)) \Leftrightarrow \varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma)$$

Preuve

($\Leftarrow$) Soit $\varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma)$. Par définition, pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$, l'ensemble des états de $[\![\mathcal{M}]\!]$ qui valident φ est soit vide soit égale à $[\![\mathcal{M}]\!]$. Par définition de la négation, ceci signifie que l'ensemble des états qui valident $\neg\varphi$ est soit égale à $[\![\mathcal{M}]\!]$ soit vide. Il est alors clair que pour tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$, soit $\mathcal{M} \models_\Sigma \varphi$ soit $\mathcal{M} \models_\Sigma \neg\varphi$, *i.e.* $\mathcal{M}od(\neg\varphi) = \mathcal{M}od(\Sigma) \setminus \mathcal{M}od(\varphi)$.

($\Rightarrow$) Soit $\varphi \in \mathcal{S}en(\Sigma)$ tel que $\mathcal{M}od(\neg\varphi) = \mathcal{M}od(\Sigma) \setminus \mathcal{M}od(\varphi)$ et soit $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$. Puisque par hypothèse $\mathcal{M}od(\Sigma) = \mathcal{M}od(\neg\varphi) \cup \mathcal{M}od(\varphi)$, on a soit $\mathcal{M} \models_\Sigma \varphi$, soit $\mathcal{M} \not\models_\Sigma \varphi$. Par définition, ceci signifie que soit pour tout $\eta \in [\![\mathcal{M}]\!]_\Sigma$ on a $\mathcal{M} \models_{\Sigma,\eta} \varphi$, soit pour tout $\eta \in [\![\mathcal{M}]\!]_\Sigma$ on a $\mathcal{M} \not\models_{\Sigma,\eta} \varphi$. On a ainsi montré que $\varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma)$.

□

On redéfinit alors en termes de formules sémantiquement fermées les notions introduites à la définition 1.3.1, partie I.

Définition 1.2.5 *Soit $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ une institution stratifiée. Soit $\Sigma \in |\mathcal{S}ig|$ et soient $\Phi \subseteq \mathcal{S}en(\Sigma)$ un ensemble de Σ-formules et $\mathscr{M} \subseteq |\mathcal{M}od(\Sigma)|$ une collection de Σ-modèles.*

1. $\mathcal{T}h(\mathscr{M})$ *(la théorie de $\mathscr{M}$) est la* collection de toutes les Σ-formules sémantiquement fermées satisfaites par tous les Σ-modèles appartenant à $\mathscr{M}$*, i.e. :*
$$\mathcal{T}h(\mathscr{M}) = \{\varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma) / \forall \mathcal{M} \in \mathscr{M},\ \mathcal{M} \models_\Sigma \varphi\}$$
2. $\mathcal{T}h(\Phi)$ *(la fermeture de Φ) est la* collection de toutes les Σ-formules sémantiquement fermées satisfaites par tous les Σ-modèles de Φ*, i.e. :*
$$\mathcal{T}h(\Phi) = \{\varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma) / \forall \mathcal{M} \in |\mathcal{M}od(\Phi)|,\ \mathcal{M} \models_\Sigma \varphi\} = \mathcal{T}h(\mathcal{M}od(\Phi))$$
3. $\models_\Sigma$ *est maintenant une relation définie sur $\wp(\mathcal{S}en(\Sigma)) \times \mathcal{S}en_{\mathcal{F}}(\Sigma)$ de la façon suivante :*
$$\forall \varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma),\ \Phi \models_\Sigma \varphi \Leftrightarrow \varphi \in \mathcal{T}h(\Phi)$$
On dit alors que φ est une conséquence sémantique (fermée) *de Φ.*
4. *Si $\Phi = \mathcal{T}h(\Phi)$ alors Φ est dit* (sémantiquement) fermé *ou* (sémantiquement) clos.
5. *Une Σ-*théorie *T est un ensemble (sémantiquement) fermé de Σ-formules, i.e. $T = \mathcal{T}h(T)$. On dit que Φ* présente *la théorie T si $\Phi \subseteq \mathcal{S}en_{\mathcal{F}}(\Sigma)$ et T est la théorie obtenue par fermeture de Φ, i.e. si $T = \mathcal{T}h(\Phi)$.*

Il en découle alors directement que deux Σ-modèles sont *élémentairement équivalents* s'ils valident le même ensemble de formules sémantiquement fermées.

Chapitre 2

Méthode des diagrammes

Nous présentons dans ce chapitre une généralisation institution-indépendante d'une méthode très importante de la théorie standard des modèles qui est la base de nombreux résultats dont le lemme de consistance de Robinson. Cette méthode, connue sous le nom de méthode des diagrammes, a déjà reçu une généralisation institution-indépendante par R. Diaconescu (*cf.* [40]). Ici, nous proposons une généralisation plus « finie » de cette méthode pour ne considérer que les diagrammes complets d'un modèle. En effet, seule l'utilisation de ces diagrammes permet d'obtenir le résultat qui nous intéresse : la consistance de Robinson. Notre généralisation des diagrammes complets a été rendu possible en les définissant non pas dans le cadre des institutions mais des institutions stratifiées. Comme nous le verrons dans ce chapitre, un autre intérêt de notre approche a été de pouvoir définir la notion de morphismes élémentaires indépendamment de celle des ciagrammes complets comme c'est le cas dans [66]. Comme nous le verrons dans ce chapitre et le suivant, ceci a permis de généraliser des résultats de théorie standard des modèles tels que l'union des chaînes de Tarski ou le théorème de Löwenheim-Skolem descendant à des logiques non-standard comme les logiques modales.

Le plan de ce chapitre est le suivant. Dans la première section nous introduisons la méthode des diagrammes telle qu'elle a été abordée dans les travaux de R. Diaconescu (*cf.* [40]). Nous présentons également les travaux de D. Găină et A. Popescu (*cf.* [66]) sur les notions de diagramme et de morphisme élémentaire. Nous présentons ensuite notre approche de la méthode des diagrammes dans le cadre des institutions stratifiées dans la deuxième section. Enfin, nous donnons en troisième section deux applications directes de la méthode des diagrammes : les théorèmes de Löwenheim-Skolem et le test de Lòs-Vaught.

2.1 Les diagrammes en théorie standard des modèles

2.1.1 Diagrammes élémentaires

La notion de diagramme élémentaire a été introduite par R. Diaconescu dans [40] en vue de remonter la méthode des diagrammes au niveau des institu-

tions.[1]. Il a alors pu montrer d'importants résultats de libéralité et d'axiomatisation, illustrant ainsi le pouvoir de cette méthode.
Afin de donner une intuition de la méthode des diagrammes, nous allons prendre l'exemple de la logique du premier ordre avec égalité pour laquelle cette méthode a été définie en premier lieu.

Exemple 2.1.1 *Étant donné un modèle du premier ordre $\mathcal{M}$, il est possible de considérer l'extension de sa signature par l'ajout des éléments de M comme nouvelles constantes. Ces nouveaux symboles seront alors interprétés canoniquement par leur propre valeur. L'ensemble $E_{\mathcal{M}}$ des atomes (positifs) satisfait par $\mathcal{M}$ dans cette signature étendue est appelé le diagramme « positif » de $\mathcal{M}$ et constitue une importante description syntaxique du modèle $\mathcal{M}$. De tels diagramme sont à la base de la méthode dite « des diagrammes » qui joue un rôle important en théorie standard des modèles.*
Si on regarde ces diagrammes d'un point de vue plus structurel, on remarque que chaque modèle $\mathcal{M}$ d'une signature $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ défini une extension de signatures $\iota : (\mathcal{S}, \mathcal{F}, \mathcal{R}) \hookrightarrow (\mathcal{S}, \mathcal{F}_{\mathcal{M}}, \mathcal{R})$ telle que :

- *$(\mathcal{F}_{\mathcal{M}})_{\omega \to s} = \mathcal{F}_{\omega \to s}$ pour tout arité non-vide $\omega \in \mathcal{S}^*$ et toute sorte $s \in \mathcal{S}$;*
- *$(\mathcal{F}_{\mathcal{M}})_{\to s} = \mathcal{F}_{\to s} \cup M_s$ pour toute sorte $s \in \mathcal{S}$.*

Le $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-modèle $\mathcal{M}$ peut être canoniquement étendu en un $(\mathcal{S}, \mathcal{F}_{\mathcal{M}}, \mathcal{R})$-modèle $\mathcal{M}_{\mathcal{M}}$ en interprétant les nouvelles constantes de $(\mathcal{F}_{\mathcal{M}})_{\to s}$ par les éléments correspondants de M_s, i.e. $a^{\mathcal{M}_{\mathcal{M}}} = a$ pour tout $a \in M$. L'ensemble $E_{\mathcal{M}}$ est l'ensemble de tous les atomes (équationnels et relationnels) satisfait par $\mathcal{M}_{\mathcal{M}}$.
La présentation $((\mathcal{S}, \mathcal{F}_{\mathcal{M}}, \mathcal{R}), E_{\mathcal{M}})$ possède la propriété catégorielle cruciale d'axiomatiser la classe des morphismes de Σ-modèles de source $\mathcal{M}$. Plus précisement, il existe un isomorphisme naturel $i : \mathcal{M}od((\mathcal{S}, \mathcal{F}_{\mathcal{M}}, \mathcal{R}), E_{\mathcal{M}}) \to \mathcal{M}/\mathcal{M}od(\mathcal{S}, \mathcal{F}, \mathcal{R})$. Cet isomorphisme associe à tout $(\mathcal{S}, \mathcal{F}_{\mathcal{M}}, \mathcal{R})$-modèle $\mathcal{N}$ satisfaisant $E_{\mathcal{M}}$ le morphisme de $(\mathcal{S}, \mathcal{F}, \mathcal{R})$-modèles $h_{\mathcal{N}} : \mathcal{M} \to \mathcal{M}od(\iota)(\mathcal{N})$ tel que $h_{\mathcal{N}}(a) = a^{\mathcal{N}}$ pour tout élément $a \in M$.

Définition 2.1.1 (diagramme élémentaire ([40])) *Soit une institution $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$. Soient $\Sigma \in |\mathcal{S}ig|$ une signature et $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ un Σ-modèle. Un* diagramme élémentaire *de $\mathcal{M}$ est la donnée d'un triplet $(\iota_\Sigma(\mathcal{M}), E_{\mathcal{M}}, i_{\Sigma,\mathcal{M}})$ tel que :*

- *$\iota_\Sigma(\mathcal{M}) : \Sigma \to \Sigma_{\mathcal{M}}$ est un morphisme de signatures, appelé* extension élémentaire de Σ via $\mathcal{M}$ *;*
- *$E_{\mathcal{M}} \in \mathcal{S}en(\Sigma_{\mathcal{M}})$ est un ensemble de $\Sigma_{\mathcal{M}}$-formules, appelé* diagramme élémentaire *du modèle $\mathcal{M}$;*
- *$i_{\Sigma,\mathcal{M}} : \mathcal{M}od((\Sigma_{\mathcal{M}}, E_{\mathcal{M}})) \to \mathcal{M}/\mathcal{M}od(\Sigma)$ est un isomorphisme*

tel que le diagramme suivant commute

$$\begin{tikzcd}
\mathcal{M}od((\Sigma_{\mathcal{M}}, E_{\mathcal{M}})) \arrow[r, "i_{\Sigma,\mathcal{M}}"] \arrow[dr, "\mathcal{M}od(\iota_\Sigma(\mathcal{M}))"'] & (\mathcal{M}/\mathcal{M}od(\Sigma)) \arrow[d, "\text{oubli}"] \\
 & \mathcal{M}od(\Sigma)
\end{tikzcd}$$

1. En fait, la notion de diagramme élémentaire définie dans [40] est la généralisation de la notion de diagramme positif en logique du premier ordre classique.

et que les deux conditions suivantes sont vérifiées :

1. *le morphisme ι est* fonctoriel, *i.e. pour tout morphisme de signature $\sigma : \Sigma \to \Sigma'$ et tout morphisme de Σ-modèles $h : \mathcal{M} \to \mathit{Mod}(\sigma)(\mathcal{M}')$ il existe un morphisme de présentations $\iota_\sigma(h) : (\Sigma_\mathcal{M}, E_\mathcal{M}) \to (\Sigma'_{\mathcal{M}'}, E_{\mathcal{M}'})$ tel que le diagramme suivant commute,*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\iota_\Sigma(\mathcal{M})} & \Sigma_\mathcal{M} \\ {\scriptstyle\sigma}\downarrow & & \downarrow{\scriptstyle\iota_\sigma(h)} \\ \Sigma' & \xrightarrow[\iota_{\Sigma'}(\mathcal{M}')]{} & \Sigma'_{\mathcal{M}'} \end{array}$$

2. *l'isomorphisme i est* naturel, *i.e. pour tout morphisme de signature $\sigma : \Sigma \to \Sigma'$ et tout homomorphisme de Σ-modèles $h : \mathcal{M} \to \mathit{Mod}(\sigma)(\mathcal{M}')$, le diagramme suivant commute.*

$$\begin{array}{ccc} \mathit{Mod}((\Sigma_\mathcal{M}, E_\mathcal{M})) & \xrightarrow{i_{\Sigma,\mathcal{M}}} & \mathcal{M}/\mathit{Mod}(\Sigma) \\ {\scriptstyle Mod(\iota_\sigma(h))}\uparrow & & \uparrow{\scriptstyle h/\mathit{Mod}(\sigma)=\mathit{Mod}(\sigma)\circ h} \\ \mathit{Mod}((\Sigma'_{\mathcal{M}'}, E_{\mathcal{M}'})) & \xrightarrow[i_{\Sigma',\mathcal{M}'}]{} & \mathcal{M}'/\mathit{Mod}(\Sigma') \end{array}$$

Intuitivement, la définition ci-dessus dit que pour tout Σ-modèle $\mathcal{M}$, la classe des morphismes de Σ-modèles de source $\mathcal{M}$ peut être axiomatisée comme la classe des modèles d'une certaine présentation : le diagramme de $\mathcal{M}$. Autrement dit, le diagramme élémentaire de $\mathcal{M}$ est la présentation encodant la classe des morphismes de Σ-modèles de source $\mathcal{M}$. Ainsi dans le cadre de la logique du premier ordre il existe, selon les morphismes de modèles considérés, différentes possibilité de diagrammes élémentaires que nous résumons dans le tableau ci-dessous.
On rappelle qu'un morphisme de Σ-modèles du premier ordre $h : \mathcal{M} \to \mathcal{N}$ est :

- *fermé* si $r^\mathcal{M} = h^{-1}(r^\mathcal{N})$ pour tout symbole de prédicat appartenant à la signature Σ ;
- une *inclusion élémentaire* si $\mathcal{M}_\mathcal{M}$ est élémentairement équivalent à $i^{-1}(h)$.

Morphismes	$E_\mathcal{M}$
tous	tous les atomes satisfaits par $\mathcal{M}_\mathcal{M}$
injectifs	tous les atomes et les négations d'équations satisfaits par $\mathcal{M}_\mathcal{M}$
fermés	tous les atomes et les négations des atomes relationnels satisfaits par $\mathcal{M}_\mathcal{M}$
fermés et injectifs	tous les atomes et leurs négations satisfaits par $\mathcal{M}_\mathcal{M}$
inclusions élémentaires	toutes les formules satisfaites par $\mathcal{M}_\mathcal{M}$

Définition 2.1.2 ([40]) *Une institution $\mathcal{I} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, \models)$ possède des diagrammes élémentaires si pour toute signature $\Sigma \in |\mathit{Sig}|$ et tout Σ-modèle $\mathcal{M}$, il existe un diagramme élémentaire de $\mathcal{M}$ au sens de la définition 2.1.1.*

2.1.2 Morphismes élémentaires

Intuitivement, une extension élémentaire en logique classique est un morphisme de modèles préservant toutes les propriétés exprimable dans le langage. Lorsqu'on se place dans le cadre général des institutions, il n'existe pas de moyens directs de « paramétriser » le langage de façon à définir les extensions élémentaires. Selon D. Găină et A. Popescu dans [66], la notion de diagramme élémentaire définie dans la section précédente offre un cadre adéquat pour y généraliser cette notion d'extension élémentaire. De fait, les morphismes élémentaires définis dans [66] viennent après et sont définis sur les diagrammes élémentaires. Nous verrons plus loin (*cf.* section 2.2) que cela n'est pas sans conséquences.

Définition 2.1.3 (morphisme élémentaire [66]) *Soit $\mathcal{I} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, \models)$ une institution. Soit $\Sigma \in |\mathcal{S}ig|$ une signature et $\mu : \mathcal{M} \to \mathcal{M}'$ un morphisme de Σ-modèles. Le morphisme μ est dit* élémentaire *si $\mathcal{S}en(\iota_\Sigma(\mu))(\mathcal{T}h(\mathcal{M}_\mathcal{M}) \subseteq \mathcal{T}h(\mathcal{M}'_{\mathcal{M}'})$, où ι_Σ et $\mathcal{M}_\mathcal{M}$ sont respectivement le morphisme de signatures et le diagramme élémentaire de $\mathcal{M}$ introduit à la définition 2.1.1.*

L'intuition derrière les morphismes élémentaires est empruntée à celle des extensions élémentaires de la logique du premier ordre. Le modèle $\mathcal{M}_\mathcal{M}$ est l'extension de $\mathcal{M}$ obtenue en ajoutant comme symbole de constante les éléments de l'ensemble de base M. Le morphisme $\mu : \mathcal{M} \to \mathcal{M}'$ est élémentaire si $\mathcal{M}'$ réalise au travers de μ toutes les formules « paramétrées par les éléments de M » vraies dans $\mathcal{M}$. La proposition suivante donne les propriétés importantes des morphismes élémentaires.

Proposition 2.1.1 ([66])

1. *si $\mu : \mathcal{M} \to \mathcal{M}'$ est un morphisme élémentaire, alors $\mathcal{T}h(\mathcal{M}) \subseteq \mathcal{T}h(\mathcal{M}')$. Si, de plus, $\mathcal{I}$ possède la négation, alors $\mathcal{M} \equiv_\Sigma \mathcal{M}'$;*
2. *les morphismes élémentaires sont préservés par réduction le long des morphismes de signatures ;*
3. *pour toute signature $\Sigma \in |\mathcal{S}ig|$, les morphismes élémentaires forment une sous-catégorie de $\mathcal{M}od(\Sigma)$;*
4. *si $\mathcal{I}$ possède la négation, alors la condition de la définition 2.1.3 devient équivalente à $\mathcal{M}_\mathcal{M} \equiv \mathcal{M}od(\iota_\Sigma(\mu))(\mathcal{M}'_{\mathcal{M}'})$.*

2.2 Notre approche

Nous avons vu dans la section précédente une première généralisation de la méthode des diagrammes et, en particulier de la notion d'extension élémentaire. Cependant, cette dernière généralisation ne s'applique qu'à une classe relativement restreinte de logiques puisqu'elle présuppose l'existence de diagrammes élémentaires. Or il existe des logiques, telle que les logiques modales, pour lesquelles il n'est pas possible de définir la notion de diagrammes. Ceci découle du fait que les états, qui stratifient la relation de satisfaction, sont implicites. Il n'ont donc pas d'équivalents syntaxiques que l'on pourrait ajouter aux signatures. Le cadre des institutions stratifiées, via les notions d'état et de stratification de la relation de satisfaction, va nous permettre de définir une notion d'extension élémentaire indépendante de la notion de diagramme et aussi de généraliser de façon plus « fine » la méthode des diagrammes en donnant une généralisation à la notion de diagramme complet, notion plus fondamentale pour la généralisation de la consistance de Robinson.

2.2.1 Extensions élémentaires

Les extensions élémentaires dans les institutions stratifiées se définissent simplement de la façon suivante.

Définition 2.2.1 (extension élémentaire) *Soit $\mathcal{SI} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, [\![]\!])$ une institution stratifiée. Soient $\Sigma \in |\mathcal{Sig}|$ une signature et $\mu : \mathcal{M} \to \mathcal{M}'$ un morphisme de Σ-modèles. Le Σ-modèle $\mathcal{M}'$ est une* extension élémentaire *de $\mathcal{M}$ par rapport à μ si pour toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ et tout $\eta \in [\![\mathcal{M}]\!]_\Sigma$, on a :*

$$\mathcal{M} \models_{\Sigma,\eta} \varphi \Longleftrightarrow \mathcal{M}' \models_{\Sigma,[\![\mu]\!]_\Sigma(\eta)} \varphi$$

On notera $\mathcal{M} \prec_\mu \mathcal{M}'$ le fait que $\mathcal{M}'$ soit une extension élémentaire de $\mathcal{M}$ par rapport à μ. Dans ce cas, on dira que μ est un Σ-morphisme élémentaire.

Dans le cadre de la logique du premier ordre, tout Σ-morphisme $\mu : \mathcal{M} \to \mathcal{M}'$ n'est pas nécessairement un morphisme élémentaire, même si c'est un monomorphisme. On a en effet le contre-exemple suivant en logique du premier ordre avec égalité.

Exemple 2.2.1 *Pour la signature des groupes, $< \mathbb{Z}, 0, + >$ est un sous-groupe de $< \mathbb{Q}, 0, + >$. Cependant, la formule $\forall x \exists y, y + y = x$ est vraie dans $\mathbb{Q}$ mais fausse dans $\mathbb{Z}$. Le monomorphisme de groupe $\mu : \mathbb{Z} \to \mathbb{Q}$ n'est donc pas élémentaire.*

En fait, un Σ-morphisme $\mu : \mathcal{M} \to \mathcal{M}'$ définit une application $[\![\mu]\!]_\Sigma : [\![\mathcal{M}]\!]_\Sigma \to [\![\mathcal{M}']\!]_\Sigma$ qui ne préserve pas nécessairement la sémantique des constructeurs abstraits de formules (formules atomiques, connecteurs logiques, quantificateurs et modalités). C'est pour cette raison que le morphisme μ n'est pas nécessairement élémentaire. Dans le chapitre 3, nous étendrons les institutions stratifiées de façon à prendre en compte cette notion de constructeur abstrait de formule et nous montrons (*cf.* proposition 3.2.1) l'équivalence des notions de morphisme élémentaire et de préservation de la sémantique des constructeurs de formules.

L'intérêt de cette généralisation de la notion d'extension élémentaire est d'être indépendante de la notion de diagramme. Bien entendu, cette définition généralise la notion d'extension élémentaire définie dans le cadre de la logique du premier ordre et, en ne supposant pas l'existence de diagrammes, capture l'essence même de la notion d'extension élémentaire. Ceci constitue la différence esentielle de notre définition par rapport à celle de D. Găină et A. Popescu présentée dans la section précédente[2].

Proposition 2.2.1 *Si $\mathcal{SI}$ est une institution stratifiée possédant la négation alors, pour tout Σ-morphisme élémentaire $\mu : \mathcal{M} \to \mathcal{M}'$, on a $\mathcal{M} \equiv_\Sigma \mathcal{M}'$.*

2. Une autre différence est que D. Găină et A. Popescu considèrent une inclusion dans la préservation des propriétés. Ceci correspond dans notre définition à ne considérer que l'implication $\mathcal{M} \models_{\Sigma,\eta} \varphi \Rightarrow \mathcal{M}' \models_{\Sigma,[\![\mu]\!](\eta)} \varphi$ au lieu de l'équivalence.

Preuve Soit $\varphi \in \mathcal{S}en_{\mathcal{F}}(\Sigma)$ une Σ-formule sémantiquement fermée telle que $\mathcal{M} \models_\Sigma \varphi$ (une telle formule si $\mathcal{S}en(\Sigma) \neq \varnothing$ existe puisque $\mathcal{I}$ est fermée par négation). Par définition, ceci signifie que pour tout état $\eta \in [\![\mathcal{M}]\!]_\Sigma$, on a $\mathcal{M} \models_{\Sigma,\eta} \varphi$. Puisque μ est une extension élémentaire, ceci est équivalent à $\mathcal{M}' \models_{\Sigma,[\![\mu]\!]_\Sigma(\eta)} \varphi$. Puisque φ est sémantiquement fermée on en déduit $\{\eta' \in [\![\mathcal{M}']\!]_\Sigma / \mathcal{M}' \models_{\Sigma,\eta'} \varphi\} = [\![\mathcal{M}']\!]_\Sigma$, *i.e.* $\mathcal{M}' \models_\Sigma \varphi$. □

Proposition 2.2.2 *Soit $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ une institution stratifiée. Pour toute signature $\Sigma \in |\mathcal{S}ig|$, les Σ-morphismes élémentaires forment une sous-catégorie $\mathcal{E}lem(\Sigma)$ de $\mathcal{M}od(\Sigma)$.*

Preuve

Identité : les morphismes identité sont évidemment des morphismes élémentaires ;

Composition : Soient $\mu : \mathcal{M} \to \mathcal{M}'$ et $\mu' : \mathcal{M}' \to \mathcal{M}''$ deux Σ-morphismes élémentaires. Il est clair d'après les définitions que $\mu' \circ \mu$ est un morphisme de Σ-modèles et que de plus $\mathcal{M} \prec_{\mu' \circ \mu} \mathcal{M}''$.

□

Proposition 2.2.3 *Soit $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ une institution stratifiée telle que pour tout morphisme de signatures $\sigma \in \mathcal{S}ig(\Sigma, \Sigma')$ et tout Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$ il existe une transformation naturelle $[\![]\!]_\sigma : [\![]\!]_{\Sigma'} \Rightarrow [\![]\!]_\Sigma \circ \mathcal{M}od(\sigma)$ satisfaisant les deux conditions suivantes :*

1. *pour tout Σ'-modèle $\mathcal{M}' \in |\mathcal{M}od(\Sigma')|$, $[\![\mathcal{M}']\!]_\sigma$ est une application surjective ;*
2. *pour toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$ et tout $\eta' \in [\![\mathcal{M}']\!]_{\Sigma'}$:*

$$\mathcal{M}' \models_{\Sigma',\eta'} \mathcal{S}en(\sigma)(\varphi) \Longleftrightarrow \mathcal{M}od(\sigma)(\mathcal{M}') \models_{\Sigma,[\![\mathcal{M}']\!]_\sigma(\eta')} \varphi$$

Alors, pour tout morphisme de signatures $\sigma : \Sigma \to \Sigma'$, on peut restreindre le foncteur $\mathcal{M}od(\sigma) : \mathcal{M}od(\Sigma') \to \mathcal{M}od(\Sigma)$ à la sous-catégorie $\mathcal{E}lem(\Sigma')$ des Σ'-morphismes élémentaires, i.e. $\mathcal{M}od(\sigma) : \mathcal{E}lem(\Sigma') \to \mathcal{E}lem(\Sigma)$. On dit alors que $\mathcal{M}od$ préserve *les morphismes élémentaires.*

Preuve Soit $\mathcal{M} \prec_\mu \mathcal{M}'$ un Σ'-morphisme élémentaire, *i.e.* $\mu \in |\mathcal{E}lem(\Sigma')|$. Soient $\eta \in [\![\mathcal{M}od(\sigma)(\mathcal{M})]\!]_\Sigma$ et $\varphi \in \mathcal{S}en(\Sigma)$. Puisque $[\![\mathcal{M}]\!]_\sigma$ est surjective, il existe un état $\eta' \in [\![\mathcal{M}]\!]_{\Sigma'}$ tel que $[\![\mathcal{M}]\!]_\sigma(\eta') = \eta$. On a donc :

$$\begin{aligned}
\mathcal{M}od(\sigma)(\mathcal{M}) \models_{\Sigma,\eta} \varphi &\Longleftrightarrow \mathcal{M} \models_{\Sigma',\eta'} \mathcal{S}en(\sigma)(\varphi) \\
&\Longleftrightarrow \mathcal{M}' \models_{\Sigma',[\![\mu]\!]_{\Sigma'}(\eta')} \mathcal{S}en(\sigma)(\varphi) \\
&\Longleftrightarrow \mathcal{M}od(\sigma)(\mathcal{M}') \models_{\Sigma,[\![\mathcal{M}']\!]_\sigma([\![\mu]\!]_{\Sigma'}(\eta'))} \varphi \\
&\Longleftrightarrow \mathcal{M}od(\sigma)(\mathcal{M}') \models_{\Sigma,[\![\mathcal{M}od(\sigma)(\mu)]\!]_\Sigma(\eta)} \varphi
\end{aligned}$$

On en déduit alors que $\mathcal{M}od(\sigma)(\mathcal{M}) \prec_{\mathcal{M}od(\sigma)(\mu)} \mathcal{M}od(\sigma)(\mathcal{M}')$. □

2.2.2 Diagrammes complets

Il est très difficile de montrer qu'un modèle est une extension élémentaire d'un autre au travers d'un morphisme. En théorie standard des modèles, la *méthode des diagrammes* permet de répondre à ce problème.

Définition 2.2.2 (diagramme complet) *Soit $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!], \models)$ une institution stratifiée. Soient $\Sigma \in |\mathit{Sig}|$ une signature et $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$ un Σ-modèle. Un* diagramme complet *de $\mathcal{M}$ est la donnée d'un triplet $(\iota_\Sigma(\mathcal{M}), \alpha, i_{\Sigma,\mathcal{M}})$ tel que :*

- *$\iota_\Sigma(\mathcal{M}) : \Sigma \to \Sigma_{\mathcal{M}}$ est un morphisme de signatures ;*
- *$\alpha : \mathit{Sen}(\Sigma) \times [\![\mathcal{M}]\!]_\Sigma \to \mathit{Sen}(\Sigma_{\mathcal{M}})$ est une application ;*
- *$i_{\Sigma,\mathcal{M}} : \mathit{Mod}(D(\mathcal{M})) \to \mathcal{M}/\mathit{Mod}(\Sigma)$ avec $D(\mathcal{M}) = \{\alpha(\varphi,\eta) \mid \mathcal{M} \models_{\Sigma,\eta} \varphi\}$ est un foncteur tel que le diagramme suivant soit commutatif.*

$$\begin{tikzcd}
\mathit{Mod}(D(\mathcal{M})) \arrow[rr, "i_{\Sigma,\mathcal{M}}"] \arrow[dr, "\mathit{Mod}(\iota_\Sigma(\mathcal{M}))"'] & & \mathcal{M}/\mathit{Mod}(\Sigma) \arrow[dl, "\text{forgetful}"] \\
& \mathit{Mod}(\Sigma) &
\end{tikzcd}$$

De plus, ce triplet satisfait les deux conditions suivantes :

1. *$Id_{\mathcal{M}} \in i_{\Sigma,\mathcal{M}}(\mathit{Mod}(D(\mathcal{M})))$*
2. *$\forall \mathcal{M}' \in \mathit{Mod}(D(\mathcal{M})), \forall (\varphi,\eta) \in \mathit{Sen}(\Sigma) \times [\![\mathcal{M}]\!]_\Sigma$*

$$\mathcal{M}' \models_\Sigma \alpha(\varphi,\eta) \Longleftrightarrow \mathit{Mod}(\iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{\Sigma,[\![i_{\Sigma,\mathcal{M}}(\mathcal{M}')]\!]_\Sigma(\eta)} \varphi$$

On note $\mathcal{D}(\mathcal{M})$ le diagramme complet de $\mathcal{M}$.
On dit que $\mathcal{SI}$ possède des diagrammes complets si pour toute signature $\Sigma \in |\mathit{Sig}|$ et pour tout Σ-modèle $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$, il existe un diagramme complet de $\mathcal{M}$.

Exemple 2.2.2 (logique du 1^{er}ordre) *Par souci de simplicité, nous allons ici restreindre la présentation au cas mono-sorte.*
Soient $\Sigma = (\mathcal{F}, \mathcal{R})$ une signature du premier ordre, $\mathcal{X}$ un ensemble de variables tel que $\mathcal{X} \cap (\mathcal{F} \cup \mathcal{R}) = \varnothing$ et soit $\mathcal{M}$ une structure du premier ordre. Soit $\Sigma_{\mathcal{M}}$ la signature définie de la façon suivante :

$$\Sigma_{\mathcal{M}} = (\mathcal{F} \cup \{c^0_m / m \in M\}, \mathcal{R})$$

et soit $\alpha : \mathit{Sen}(\Sigma) \times [\![\mathcal{M}]\!]_\Sigma \to \mathit{Sen}(\Sigma_{\mathcal{M}})$ l'application qui associe à toute Σ-formule $\varphi(x_1, \ldots, x_n)$[3] *la $\Sigma_{\mathcal{M}}$-formule $\varphi(c_{\nu(x_1)}, \ldots, c_{\nu(x_n)})$ pour une interprétation $\nu \in [\![\mathcal{M}]\!]$. Alors, $\mathcal{D}(\mathcal{M})$ est l'ensemble défini de la façon suivante :*

$$\mathcal{D}(\mathcal{M}) = \{\varphi(c_{\nu(x_1)}, \ldots, c_{\nu(x_n)}) / \mathcal{M} \models_{\Sigma,\nu} \varphi(x_1, \ldots, x_n)\}$$

Soit $\mathcal{M}' \in \mathit{Mod}((\Sigma_{\mathcal{M}}, \mathcal{D}(\mathcal{M})))$. Pour tout $m \in M$, soit $\mu(m)$ l'interprétation de m dans M'. Il est alors clair que μ est une application de M dans M'. Pour

3. La notation $\varphi(x_1, \ldots, x_n)$ signifie que les variables libres de φ sont contenues dans l'ensemble $\{x_1, \ldots, x_n\}$.

toute Σ-formule $\varphi(x_1,\dots,x_n)$ et $\iota : \mathcal{X} \to M$, $\varphi(c_{\iota(x_1)},\dots,c_{\iota(x_n)}) \in \mathcal{D}(\mathcal{M})$. On a donc :

$$\mathcal{M} \models_\iota \varphi(x_1,\dots,x_n) \Longleftrightarrow \mathit{Mod}(\iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{[\![\mu]\!]_\Sigma(\iota)} \varphi(x_1,\dots,x_n)$$

De ce fait, la première implication est vérifiée par définition de $\mathcal{D}(\mathcal{M})$ et la seconde car $\mathcal{M}' \in \mathit{Mod}(\Sigma_\mathcal{M},\mathcal{D}(\mathcal{M}))$, $c_{\iota(x_i)}$ est interprété par $\mu(\iota(x_i))$ et $\mathcal{D}(\mathcal{M})$ est une théorie complète. En effet, μ est un morphisme élémentaire. On pose alors $i_{\Sigma,\mathcal{M}}(\mathcal{M}') = \mu$.

Exemple 2.2.3 (logique du 2^eordre) À vérifier ! *Par souci de simplicité, nous restreingons ici la présentation au cas mono-sorte.*
Soient $\Sigma = (\mathcal{F},\mathcal{R})$ une signature du premier ordre, $\mathcal{X}$ un ensemble de variables (d'individu) tel que $\mathcal{X} \cap (\mathcal{F} \cup \mathcal{R}) = \varnothing$, χ un ensemble de variables (de relations) tel que $\chi \cap (\mathcal{F} \cup \mathcal{R} \cup \mathcal{X}) = \varnothing$ et $\mathcal{M}$ une structure du premier ordre. Soit $\Sigma_\mathcal{M}$ la signature définie de la façon suivante :

$$\Sigma_\mathcal{M} = (\mathcal{F} \cup \{c^0_m / m \in M\}, \mathcal{R} \cup \{r_{\mathscr{M}} / \mathscr{M} \in \wp(M)\})$$

et soit $\alpha : \mathit{Sen}(\Sigma) \times [\![\mathcal{M}]\!]_\Sigma \to \mathit{Sen}(\Sigma_\mathcal{M})$ l'application qui associe à toute formule $\varphi(x_1,\dots,x_n,\chi_1,\dots,\chi_m),\iota)$[4] la $\Sigma_\mathcal{M}$-formule $\varphi(c_{\iota(x_1)},\dots,c_{\iota(x_n)},r_{\iota(\chi_1)},\dots,r_{\iota(\chi_m)})$. Alors, $\mathcal{D}(\mathcal{M})$ est l'ensemble défini de la façon suivante :

$$\mathcal{D}(\mathcal{M}) = \{\varphi(c_{\iota(x_1)},\dots,c_{\iota(x_n)},r_{\iota(\chi_1)},\dots,r_{\iota(\chi_m)}) / \mathcal{M} \models_\iota \varphi(x_1,\dots,x_n,\chi_1,\dots,\chi_m)\}$$

Soit $\mathcal{M}' \in \mathit{Mod}((\Sigma_\mathcal{M},\mathcal{D}(\mathcal{M})))$. Pour tout $m \in M$ et pour tout $\mathscr{M} \in \wp(M)$, soient $\mu(m)$ l'interprétation de m dans M' et $\mu(\mathscr{M})$ l'interprétation de $\mathscr{M}$ dans M'. Il est alors clair que μ est une application de M dans M'. Pour toute Σ-formule $\varphi(x_1,\dots,x_n)$ et $\iota : \mathcal{X} \to M$, $\varphi(c_{\iota(x_1)},\dots,c_{\iota(x_n)}) \in \mathcal{D}(\mathcal{M})$. On a donc :

$$\mathcal{M} \models_\iota \varphi(x_1,\dots,x_n,\chi_1,\dots,\chi_m)$$
$$\Longleftrightarrow$$
$$\mathit{Mod}(\iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{[\![\mu]\!]_\Sigma(\iota)} \varphi(x_1,\dots,x_n,\chi_1,\dots,\chi_m)$$

De ce fait, la première implication est vérifiée par définition de $\mathcal{D}(\mathcal{M})$ et la seconde car $\mathcal{M}' \in \mathit{Mod}(\Sigma_\mathcal{M},\mathcal{D}(\mathcal{M}))$, $c_{\iota(x_i)}$ est interprété par $\mu(\iota(x_i))$, $r_{\iota(\chi_j)}$ est interprété par $\mu(\iota(\chi_j))$ et $\mathcal{D}(\mathcal{M})$ est une théorie complète. En effet, μ est un morphisme élémentaire. On pose alors $i_{\Sigma,\mathcal{M}}(\mathcal{M}') = \mu$.

Cette définition de la notion de diagramme complet est proche de celle donnée dans [40] par R. Diaconescu et présentée à la section 2.1.1. La principale différence est que lorsque l'on suppose la négation, la réduction de modèles le long du morphisme de signatures $\iota_\Sigma(\mathcal{M})$ de tout Σ-modèle $\mathcal{M}' \in \mathit{Mod}(\mathcal{D}(\mathcal{M}))$ est une extension élémentaire de $\mathcal{M}$ et donc $\mathcal{M} \equiv_\Sigma \mathit{Mod}(\iota_\Sigma(\mathcal{M}))(\mathcal{M}')$. Ceci vient du fait que les diagrammes considérés sont complets.

Théorème 2.2.1 *Si $\mathcal{SI} = (\mathit{Sig},\mathit{Sen},\mathit{Mod},[\![]\!],\models)$ est une institution stratifiée possédant la négation alors, pour toute signature $\Sigma \in |\mathit{Sig}|$ et tout Σ-modèle $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$, $i_{\Sigma,\mathcal{M}}$ peut être co-restreint à $\mathcal{M}/\mathit{Elem}(\Sigma)$.*

4. La notation $\varphi(x_1,\dots,x_n,\chi_1,\dots,\chi_m)$ signifie que les variables libres de φ sont contenues dans l'ensemble $\{x_1,\dots,x_n\} \cup \{\chi_1,\dots,\chi_m\}$.

Preuve Il suffit de montrer que pour tout modèle $\mathcal{M}' \in \mathcal{M}od((\Sigma_{\mathcal{M}}, D(\mathcal{M})))$, le morphisme $i_{\Sigma,\mathcal{M}}(\mathcal{M}')$ est un morphisme élémentaire, *i.e.* pour toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$ et tout $\eta \in [\![\mathcal{M}]\!]_\Sigma$, on a :

$$\mathcal{M} \models_{\Sigma,\eta} \varphi \Longleftrightarrow \mathcal{M}od(\iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{\Sigma,[\![i_{\Sigma,\mathcal{M}}(\mathcal{M}')]\!]_\Sigma(\eta)} \varphi$$

($\Rightarrow$) Supposons $\mathcal{M} \models_{\Sigma,\eta} \varphi$. Ceci signifie que $\alpha(\varphi,\eta) \in D(\mathcal{M})$ et donc $\mathcal{M}' \models_{\Sigma_{\mathcal{M}}} \alpha(\varphi,\eta)$. Par le second point de la définition 2.2.2, on en conclut alors :

$$\mathcal{M}od(\iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{\Sigma,[\![\iota_{\Sigma,\mathcal{M}}(\mathcal{M}')]\!]_\Sigma(\eta)} \varphi$$

($\Leftarrow$) (Par l'absurde)
Supposons $\mathcal{M}' \models_{\Sigma_{\mathcal{M}}} \alpha(\varphi,\eta)$ et $\mathcal{M} \not\models_{\Sigma,\eta} \varphi$. Ceci signifie que $\alpha(\varphi,\eta) \notin D(\mathcal{M})$. Puisque $\mathcal{SI}$ possède la négation, $\mathcal{D}(\mathcal{M})$ est une théorie complète et donc $\neg\alpha(\varphi,\eta) \in \mathcal{D}(\mathcal{M})$. On en déduit alors $\mathcal{M}' \models_{\Sigma_{\mathcal{M}}} \neg\alpha(\varphi,\eta)$ ce qui contredit les hypothèses.

$\square$

De par la définition 2.2.2, la notion de diagramme complet n'a de sens que dans la mesure où il est possible de donner un équivalent syntaxique aux objets utilisés pour stratifier la satisfaction des formules. De ce fait, la méthode des diagrammes ne peut, comme nous l'avons déjà signalé dans la section précédente, être utilisée pour les logiques modales. En effet, étant données une signature modale Σ et un Σ-modèle $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$, il n'existe pas de moyen d'étendre la signature Σ en une signature $\Sigma_{\mathcal{M}}$ par des objets syntaxiques tels que des constantes de façon à ce que pour toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$ et tout état $\eta \in \mathfrak{E}$ il existe une $\Sigma_{\mathcal{M}}$-formule ψ telle que : $\mathcal{M} \models_{\Sigma,\eta} \varphi \Longleftrightarrow i_{\Sigma,\mathcal{M}}(Id_{\mathcal{M}}) \models_{\Sigma_{\mathcal{M}}} \psi$. Cela est du au fait que les états ne peuvent pas être représentés par des éléments de la signature, *i.e.* des symboles dont les interprétations seraient les états d'un modèle. La raison pour cela est que les états son implicites en logique modale.

2.3 Application

Nous donnons ici une application bien connue en logique du premier ordre classique de la méthode des diagrammes : les théorèmes de Löwenheim-Skolem. Nous généralisons également un autre résultat important de la théorie standard des modèles, conséquence directe du théorème de Löwenheim-Skolem : le test de Lòs-Vaught.

2.3.1 Les deux théorèmes de Löwenheim et Skolem

Pour établir les deux théorèmes de Löwenheim-Skolem, nous avons auparavant besoin de définir la notion de cardinal d'une signature et d'un modèle. La convention que nous adoptons est la suivante : le cardinal d'une signature Σ est le cardinal de l'ensemble $\mathcal{S}en(\Sigma)$ des Σ-formules et le cardinal d'un Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ est le cardinal de l'ensemble $[\![\mathcal{M}]\!]_\Sigma$.

Théorème descendant

Pour établir une preuve institution-indépendante du théorème de Löwenheim-Skolem descendant, nous avons besoin des notions suivantes :

Définition 2.3.1 (extension libre) *Soit $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!])$ une institution stratifiée. On dit que $\mathcal{SI}$ possède des* extensions libres *si pour toute signature $\Sigma \in |\mathit{Sig}|$, tout Σ-modèle $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$ et tout $A \subseteq [\![\mathcal{M}]\!]_\Sigma$, il existe un Σ-modèle $J(A) \in |\mathit{Mod}(\Sigma)|$ et une application $f_A : A \to [\![J(A)]\!]_\Sigma$ telle que pour tout $\mathcal{M}' \in |\mathit{Mod}(\Sigma)|$ et tout $g : A \to [\![\mathcal{M}']\!]_\Sigma$ il y a un unique $\mu_g : J(A) \to \mathcal{M}'$ tel que le diagramme suivant commute.*

$$\begin{tikzcd} A \arrow[r, "f_A"] \arrow[d, "g"'] & {[\![J(A)]\!]_\Sigma} \arrow[dl, "{[\![\mu_g]\!]_\Sigma}"] \\ {[\![\mathcal{M}']\!]_\Sigma} & \end{tikzcd}$$

Proposition 2.3.1 *Soit $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!], \models)$ une institution stratifiée possédant des extensions libres. Soient $\Sigma \in |\mathit{Sig}|$ une signature et $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$ un Σ-modèle. Soit $\mathcal{C}$ la catégorie associée à l'ensemble partiellement ordonné $(\wp([\![\mathcal{M}]\!]_\Sigma), \subseteq)$. Soit $J : \mathcal{C} \to \mathit{Mod}(\Sigma)$ le foncteur qui associe à tout $A \subseteq [\![\mathcal{M}]\!]_\Sigma$ le Σ-modèle $J(A)$ et à toute inclusion $A \subseteq B$ le morphisme $\mu_{f_B \circ A \subseteq B}$. Alors, $(\mathcal{M}, \mu)$ est la colimite de J où $\mu_{J(A)} = \mu_{A \subseteq [\![\mathcal{M}]\!]_\Sigma}$.*

Preuve L'universalité du cône $(\mathcal{M}, \mu)$ de sommet le Σ-modèle $\mathcal{M}$ et de base le foncteur J découle directement de la commutativié du diagramme d'extension libre ci-dessus. $\square$

Dans la suite, toute institution stratitfiée $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!], \models)$ que nous considèrerons sera supposée vérifier les propriétés suivantes :

- $\mathcal{SI}$ possède des extensions libres ;
- pour toute signature $\Sigma \in |\mathit{Sig}|$, tout Σ-modèle $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$ et tout ensemble $A \subseteq [\![\mathcal{M}]\!]_\Sigma$,
  1. $[\![\mu_{J(A)}]\!]_\Sigma$ est injective,
  2. si $\mathit{card}(A) \geq \mathit{card}(\Sigma)$ alors $\mathit{card}(J(A)) = \mathit{card}(A)$,
  3. pour toute Σ-formule $\varphi \in \mathit{Sen}(\Sigma)$ et tout $\eta \in [\![\mu_{J(A)}]\!]_\Sigma(A)$ il existe un ensemble fini $A_{(\varphi,\eta)} \subseteq [\![\mathcal{M}]\!]_\Sigma$ tel que :
     (a) $\mathcal{M} \models_{\Sigma,\eta} \varphi \Longleftrightarrow J(B) \models_{\Sigma,[\![\mu_{J(B)}]\!]_\Sigma^{-1}(\eta)} \varphi$
     (b) $J(A) \prec_{J(A \subseteq B)} J(B)$

     où $B = A \cup A_{(\varphi,\eta)}$

Pour les logiques possédant des extensions libres, $J(A)$, pour $A \subseteq [\![\mathcal{M}]\!]_\Sigma$, dénote le plus petit Σ-modèle engendré par A. En logique du premier ordre, étant donnée une Σ-structure $\mathcal{M}$ et un ensemble $A \subseteq M^{\mathcal{X}}$, $J(A)$ est la Σ-structure engendrée par l'ensemble $B = \bigcup_{\nu \in A} \bigcup_{x \in \mathcal{X}} \nu(x)$, *i.e.* le plus petit sous-ensemble M' de M contenant B et fermé par les fonctions. $J(A)$ est une sous-structure de $\mathcal{M}$. De plus, elle a la même cardinalité que B et donc que A (toute valeur dans M est le résultat de l'interprétation d'un terme avec variable dans B et l'ensemble de ces termes est l'ensemble de séquences finies de $F \times B$ qui est donc de cardinalité plus faible que B).
En logique propositionnelle modale, étant donné un modèle de Kripke $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$,

le foncteur J asocie à tout sous-ensemble $\mathfrak{E}' \subseteq \mathfrak{E}$ le modèle de Kripke $(\mathfrak{E}', \mathfrak{R}_{|\mathfrak{E}'}, \mathfrak{v}|\mathfrak{E}')$.

Le second point ci-dessus généralise le test de Tarski-Vaught. En logique du premier ordre, pour toute formule φ et toute interprétation des variables $\nu : \mathcal{X} \to M$ telle que $\mathcal{M} \models_\nu \varphi$, on définit l'ensemble $A_{(\varphi,\nu)}$ sur la structure de φ comme suit :

- $A_{(p(t_1,\ldots,t_n),\nu)} = \emptyset$
- $A_{(\neg\psi,\nu)} = A_{(\psi,\nu)}$
- $A_{(\psi\vee\xi,\nu)} = A_{(\psi,\nu)} \cup A_{(\xi,\nu)}$
- $A_{(\exists x.\psi,\nu)} = A_{(\psi,\nu')}$ pour un $\nu' : \mathcal{X} \to M'$ quelconque vérifiant $y \neq x \in \mathcal{X}$ $\nu'(y) = \nu(y)$ et tel que $\mathcal{M}' \models_{\nu'} \psi$.

En logique propositionnelle modale, pour toute formule φ et tout état $\eta \in \mathfrak{E}'$ tel que $\mathfrak{E}' \models_\eta \varphi$, on définit inductivement l'ensemble $A_{(\varphi,s)}$ sur la structure de φ de la façon suivante :

- $A_{(p,\eta)} = \emptyset$
- $A_{(\neg\psi,\eta)} = A_{(\psi,\eta)}$
- $A_{(\psi\vee\xi,\eta)} = A_{(\psi,\eta)} \cup A_{(\xi,\eta)}$
- $A_{(@\psi,\eta)} = A_{(\psi,\eta')}$ où $@ \in \{\Box, \Diamond\}$ pour un $\eta' \in \mathfrak{E}'$ quelconque tel que $\eta\mathfrak{R}'\eta'$ et $\mathcal{M}' \models_{\eta'} \psi$.

Dans les deux cas, en supposant $A \subseteq [\![\mathcal{M}]\!]_\Sigma$ tel que $\eta \in A$, par induction mathématique sur la structure des formules, on montre les conditions 3.a et 3.b ci-dessus.

Théorème 2.3.1 (théorème de Löwenheim-Skolem descendant) *Soient $\Sigma \in |\mathcal{Sig}|$ une signature et $\mathcal{M}' \in |\mathcal{Mod}(\Sigma)|$ un Σ-modèle. Soit κ un cardinal tel que $card(\Sigma) \leq \kappa \leq card(\mathcal{M}')$. Alors, il existe un morphisme $\mu : \mathcal{M} \to \mathcal{M}'$ tel que $card(\mathcal{M}) = \kappa$.*

Preuve Soit $X \subseteq [\![\mathcal{M}']\!]_\Sigma$ tel que $card(X) = \kappa$. Soit $A_0 \subseteq A_1 \subseteq \ldots \subseteq A_n \subseteq \ldots \in \wp([\![\mathcal{M}']\!]_\Sigma)$ la séquence de sous-ensembles de $[\![\mathcal{M}']\!]_\Sigma$ définie par induction sur ω de la façon suivante :

- $A_0 = X$;
- A_{i+1} est défini à partir de A_i de la façon suivante :

$$A_{i+1} = A_i \cup \bigcup_{\substack{(\varphi,\eta) \in \mathcal{Sen}(\Sigma) \times [\![J(A_i)]\!]_\Sigma, \\ J(A_i) \not\models_{\Sigma,\eta} \varphi \\ \wedge \\ \mathcal{M}' \models_{\Sigma, [\![\mu_{J(A_i)}]\!]_\Sigma(\eta)} \varphi}} A_{(\varphi, [\![\mu_{J(A_i)}]\!]_\Sigma(\eta))}$$

Soit $B = \bigcup_{i<\omega} A_i$. Par définition, il existe $card(\Sigma)$ formules φ et $card(A_i)$ états dans $[\![J(A_i)]\!]_\Sigma$ pour obtenir A_{i+1}, ce qui montre que $card(A_{i+1}) = card(A_i) = card(X) = \kappa$. On a donc $card(J(B)) = \kappa$.
Montrons maintenant que $\mu_{J(B)}$ est un morphisme élémentaire. Soit $\mathcal{M} \models_{\Sigma,[\![\mu_{J(B)}]\!]_\Sigma(\eta)} \varphi$. Ceci signifie qu'il existe $i \in \mathbb{N}$ tel que $\eta' =$

$[\![\mu_{J(B)}]\!]_\Sigma(\eta) \in A_i$. De ce fait, soit on a $J(A_i) \models_{\Sigma,[\![\mu_{J(A_i)}]\!]_\Sigma^{-1}(\eta')} \varphi$ soit, par construction de A_{i+1} et par le point $3.a$, on a :

$$J(A_{i+1}) \models_{\Sigma,[\![J(A_i \subseteq A_{i+1})]\!]_\Sigma(\eta')} \varphi$$

et donc, par le point $3.b$, on a $J(B) \models_{\Sigma,\eta} \varphi$. L'implication inverse découle directement du point $3.b$.

□

Théorème ascendant

De même que précédemment, pour établir la preuve du théorème ascendant de Löwenheim-Skolem, nous avons besoin de conditions supplémentaires. Dans la suite, toute institution $\mathcal{SI} = (\mathcal{S}ig, \mathcal{S}en, \mathcal{M}od, [\![]\!])$ que nous considèrerons sera supposée vérifier les propriétés suivantes :

1. $\mathcal{S}ig$ possède des pushouts ;
2. $\mathcal{SI}$ est compacte, possède des diagrammes complets et la propriété d'amalgation faible pour les pushouts de morphismes de signatures ;
3. pour toute signature $\Sigma \in |\mathcal{S}ig|$ et tout cardinal κ il existe un morphisme de signatures $\sigma : \Sigma \to \Sigma_\kappa$ et une Σ_κ-theorie consistante S_κ dont le cardinal est κ et telle que pour tout sous-ensemble S de S_κ, on a :
   - $\forall \mathcal{M}' \in \mathcal{M}od(S)$, $card(\mathcal{M}') \geq card(S)$,
   - $\forall \mathcal{M} \in \mathcal{M}od(\Sigma)$,

$$card(\mathcal{M}) \geq card(S) \Leftrightarrow (\exists \mathcal{M}' \in \mathcal{M}od(S),\ \mathcal{M}od(\sigma)(\mathcal{M}') = \mathcal{M})$$

Exemple 2.3.1 *Dans le cadre de la logique du premier ordre avec égalité, étant donnée une signature* $\Sigma = (\mathcal{F}, \mathcal{R})$ *et un cardinal* κ, Σ_κ *est la signature* $(\mathcal{F}_\kappa, \mathcal{R})$ *où* $\mathcal{F}_\kappa = \mathcal{F} \cup \{c_i / i \in \kappa\}$ *et* $S_\kappa = \{\neg c_i = c_j / i, j \in \kappa \wedge i \neq j\}$.

Théorème 2.3.2 (théorème de Löwenheim-Skolem ascendant) *Soient* $\Sigma \in |\mathcal{S}ig|$ *une signature et* $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ *un* Σ*-modèle tel que* $card(\mathcal{M}) \geq \aleph_0$ *et soit* κ *un cardinal tel que* $\kappa \geq card(\Sigma)$ *et* $\kappa > card(\mathcal{M})$. *Alors, il existe un* Σ*-modèle* $\mathcal{M}' \in |\mathcal{M}od(\Sigma)|$ *et un* Σ*-morphisme* $\mu : \mathcal{M} \to \mathcal{M}'$ *tel que :*
- $\mathcal{M} \prec_\mu \mathcal{M}'$;
- $card(\mathcal{M}') = \kappa$.

Preuve Par hypothèse, il existe un morphisme de signatures $\sigma : \Sigma \to \Sigma_\kappa$ et un ensemble $S_\kappa \subseteq \mathcal{S}en(\Sigma_\kappa)$ tel que tout Σ_κ-modèles $\mathcal{M}' \in \mathcal{M}od(S_\kappa)$ possède un cardinal plus grand ou égal à κ, *i.e.* $card(\mathcal{M}') \geq \kappa$. Puisque $\mathcal{S}ig$ possède des pushouts, le pushout de signatures suivant existe donc.

$$\begin{array}{ccc} \Sigma & \xrightarrow{\iota_\Sigma(\mathcal{M})} & \Sigma_{\mathcal{M}} \\ {\scriptstyle\sigma}\downarrow & & \downarrow{\scriptstyle\rho_1} \\ \Sigma_\kappa & \xrightarrow[\rho_2]{} & \Sigma' \end{array}$$

Soit alors T' l'ensemble de Σ'-formules suivant :

$$T' = \mathcal{S}en(\rho_1)(D(\mathcal{M})) \cup \mathcal{S}en(\rho_2)(S_\kappa)$$

L'ensemble T' est consistant. En effet, puisque $\mathcal{SI}$ est compacte, tout sous-ensemble fini T de T' possède un sous-ensemble fini S de S_κ. Par définition, il existe $n \in \mathbb{N}$, un morphisme de signatures $\sigma_n : \Sigma \to \Sigma_n$ et un ensemble $S \subset \mathit{Sen}(\Sigma_n)$ satisfaisant la condition 3. Puisque $\mathit{card}(\mathcal{M}) > \aleph_0$ il existe alors $\mathcal{M}' \in \mathit{Mod}(S)$ tel que $\mathit{Mod}(\sigma_n)(\mathcal{M}') = \mathcal{M}$. Puisque $\mathcal{SI}$ possède des diagrammes élémentaires, $i^{-1}_{\Sigma,\mathcal{M}}(Id_{\mathcal{M}}) \subseteq \mathit{Mod}(D(\mathcal{M}))$. Par la propriété d'amalgation faible, T possède un Σ'-modèle. Par la propriété de compacité, il en est de même pour T'. En effet, il existe un Σ'-modèle $\mathcal{N} \in \mathit{Mod}(T')$ tel que $\mathit{Mod}(\rho_1)(\mathcal{N}) \in \mathit{Mod}(D(\mathcal{M}))$ et $\mathit{Mod}(\rho_2)(\mathcal{N}) \in S_\kappa$. On a alors $\mathit{card}(\mathit{Mod}(\rho_2)(\mathcal{N})) \geq \kappa$ et donc $\mathit{card}(\mathit{Mod}(\rho_2 \circ \sigma)(\mathcal{N})) \geq \kappa$ également.
Cependant, $\mathcal{N}$ peut avoir une taille strictement supérieure à κ. Dans ce cas, on obtient notre Σ-modèle final en utilisant le théorème descendant. □

Des deux théorèmes de Löwenheim-Skolem on a directement le corollaire suivant.

Corollaire 2.3.1 *Soient $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!], \models)$ une institution stratifiée et $\Sigma \in |\mathit{Sig}|$ une signature. Soient T une Σ-theorie et κ un cardinal tel que $\kappa \geq \mathit{card}(\Sigma)$. Si T possède un Σ-modèle $\mathcal{M} \in |\mathit{Mod}(\Sigma)|$ tel que $\mathit{card}(\mathcal{M}) > \aleph_0$ alors T possède un Σ-modèle de cardinalité κ.*

2.3.2 Le test de Lòs-Vaught

Le test de Lòs-Vaught est un théorème très important de la théorie standard des modèles. De façon à pouvoir établir ce résultat, nous avons besoin d'introduire les notions suivantes :

Définition 2.3.2 (α-catégorique) *Soient $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!])$ une institution stratifiée et $\Sigma \in |\mathit{Sig}|$ une signature. Soit T une Σ-theorie. T est dite* α-catégorique *si T possède exactement (à isomorphisme près) un Σ-modèle de cardinalité α.*

Le théorème suivant est alors une conséquence directe du théorème de Lowenheim-Skolem ascendant.

Théorème 2.3.3 (Lòs-Vaught Test) *Soit $\mathcal{SI} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, [\![]\!])$ une institution stratifiée possédant la négation et vérifiant le théorème de Löwenheim-Skolem ascendant. Soit T une Σ-théorie dont tous les modèles $\mathcal{M}$ sont tels que $\mathit{card}(\mathcal{M}) \geq \aleph_0$. Supposons de plus que T est α-catégorique pour un certain cardinal $\alpha \geq \mathit{card}(\Sigma)$. Alors, T est complète.*

Preuve Supposons le contraire. Ceci signifie qu'il existe une Σ-formule $\varphi \in \mathit{Sen}(\Sigma)$ telle que $T \cup \{\varphi\}$ et $T \cup \{\neg\varphi\}$ sont toutes deux des théories consistantes. par le théorème de Löwenheim-Skolem ascendant, il exite deux Σ-modèles $\mathcal{M}_1$ et $\mathcal{M}_2$ de cardinalité α qui sont respectivement modèle de $T \cup \{\varphi\}$ et $T \cup \{\neg\varphi\}$. Les modèles $\mathcal{M}_1$ et $\mathcal{M}_2$ ne sont alors pas isomorphes ce qui contredit l'hypothèse que T est α-catégorique. □

Chapitre 3

Institutions stratifiées avec constructeurs de formules

Dans [66], les auteurs donnent une preuve institution-indépendante du théorème des chaînes élémentaire de Tarski dans des institutions particulières appellées *institutions accessibles.* Ces dernières sont simplement des institutions du premier ordre dans le sens où les formules sont obtenues par fermeture de l'ensemble formé des abstractions de formules atomiques par les connecteurs logiques (négation et disjonction) et les quantificateurs. Nous généralisons ici ce résultat à tout constructeur de formules dans une spécialisation des institutions stratifiées appelée κ-institution stratifiées. Enfin, nous donnons aussi une preuve institution-stratifiée-indépendante du lemme de consistance de Robinson. La preuve donnée ici est une généralisation de celle donnée par Robinson dans le cadre de la théorie standard des modèles.

3.1 Définitions et exemples

Afin d'introduire les κ-institutions stratifiées, nous avons besoin de définir les notions suivantes.

Notation 3.1.1 *Soit $A \in |\mathcal{Set}|$ un ensemble. On notera 2^A l'ensemble des applications de A dans $\{0,1\}$.*

Définition 3.1.1 (signature de constructeurs) *Une* signature de constructeurs *est la donnée d'une famille $\mathcal{C} = (\mathcal{C}^n)_{n\in\mathbb{N}}$ d'ensembles indéxés par l'ensemble des entiers naturels. Pour tout $n \in \mathbb{N}$, un élément $c \in \mathcal{C}^n$ est appelé* constructeurs d'arité *n et on notera indifféremment $c \in \mathcal{C}^n$ ou $c^n \in \mathcal{C}$.*

Définition 3.1.2 (morphisme de constructeurs) *Un* morphisme de signatures de constructeurs *est la donnée d'une application $\varsigma : \mathcal{C}_1 \to \mathcal{C}_2$ préservant l'arité, i.e. :*

$$\forall n \in \mathbb{N}, \forall c \in \mathcal{C}_1^n,\ \varsigma(c) \in \mathcal{C}_2^n$$

Il est clair que les signatures de constructeurs constituent les objets d'une catégorie dont les morphismes sont les morphismes de constructeurs (*cf.* démonstration en annexe C). Dans la suite, nous noterons $\mathcal{LogSig}$ la *catégorie des signatures de constructeurs.*

Définition 3.1.3 (algèbre de constructeurs) *Soit $\mathcal{C} = (\mathcal{C}^n)_{n \in \mathbb{N}}$ une signature de constructeurs. Une* $\mathcal{C}$-algèbre *$\mathcal{A}$ est la donnée d'un ensemble de base A et, pour tout $c \in \mathcal{C}^n$, d'une application $c^{\mathcal{A}} : (2^A)^n \to 2^A$.*

Définition 3.1.4 (morphisme de $\mathcal{C}$-algèbre) *Soit $\mathcal{C} = (\mathcal{C}^n)_{n \in \mathbb{N}}$ une signature de constructeurs et soient $\mathcal{A}$ et $\mathcal{B}$ deux $\mathcal{C}$-algèbres. Un* morphisme de $\mathcal{C}$-algèbres *est la donnée d'une application $\mu : A \to B$ telle que:*
$\forall n \in \mathbb{N}, \forall c \in \mathcal{C}^n, \forall (b_1, \ldots, b_n) \in (2^B)^n, \forall \eta \in A,$

$$c^{\mathcal{B}}(b_1, \ldots, b_n)(\mu(\eta)) \quad = \quad c^{\mathcal{A}}(b_{1|A}, \ldots, b_{n|A})(\eta) \tag{3.1}$$

où, pour tout $1 \leq i \leq n$, $b_{i|A} : \begin{array}{l} A \to \{0,1\} \\ \eta \mapsto b_i(\mu(\eta)) \end{array}$ est la restriction de b_i aux éléments de A.

Il est alors clair que les $\mathcal{C}$-algèbres constituent les objets d'une catégorie dont les morphismes sont les morphismes de $\mathcal{C}$-algèbres (*cf.* démonstration en annexe C). Dans la suite, nous noterons $\mathcal{LogMod}(\mathcal{C})$ la *catégorie des $\mathcal{C}$-algèbres.*

La définition précédente indique que pour tout objet $\mathcal{C}$ de la catégorie $\mathcal{LogSig}$, la sémantique des symboles de constructeurs de formules apparaissant dans $\mathcal{C}$ est donnée par une catégorie de modèles $\mathcal{LogMod}(\mathcal{C}) \in |\mathcal{Cat}|$. Ceci nous permet de définir le foncteur contravariant $\mathcal{LogMod} : \mathcal{LogSig} \to \mathcal{Cat}$ qui, à toute signature de constructeurs $\mathcal{C} \in |\mathcal{LogSig}|$, associe la catégorie des $\mathcal{C}$-modèles $\mathcal{LogMod}(\mathcal{C})$ et, à tout morphisme de constructeurs $\varsigma \in \mathcal{LogSig}(\mathcal{C}, \mathcal{C}')$, le foncteur d'oubli $\mathcal{LogMod}(\varsigma) : \mathcal{LogMod}(\mathcal{C}') \to \mathcal{LogMod}(\mathcal{C})$ tel que, pour toute $\mathcal{C}'$-algèbre de constructeurs $\mathcal{A}' \in |\mathcal{LogMod}(\mathcal{C}')|$, $\mathcal{LogMod}(\varsigma)(\mathcal{A}') = \mathcal{A} \in |\mathcal{LogMod}(\mathcal{C})|$ est la $\mathcal{C}$-algèbre définie par $A = \mathcal{LogMod}(\varsigma)(A')$ et, pour tout symbole de constructeurs $c \in \mathcal{C}$, $c^{\mathcal{A}} = \mathcal{LogMod}(\varsigma(c)^{\mathcal{A}'})$.

On peut maintenant donner notre définition des κ-institutions stratifiées.

Définition 3.1.5 (κ-institution stratifiée) *Une* κ-institution stratifiée *$\kappa\mathcal{SI}$ est la donnée d'une institution stratifiée $\mathcal{SI} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, [\![.]\!], \models)$ munie d'un foncteur $\kappa : \mathcal{Sig} \to \mathcal{LogSig}$ et, pour toute signature $\Sigma \in |\mathcal{Sig}|$ et tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, d'une $\kappa(\Sigma)$-algèbre $\mathcal{A}$. Dans la suite, cette $\kappa(\Sigma)$-algèbre associé à $\mathcal{M}$ sera notée $[\![\mathcal{M}]\!]_\Sigma$. De plus, $\kappa\mathcal{SI}$ vérifie $\mathcal{Sen}(\Sigma) = T_{\kappa(\Sigma)}$, où $T_{\kappa(\_)} : \mathcal{Sig} \to \mathcal{Set}$ est le foncteur qui à toute signature $\Sigma \in |\mathcal{Sig}|$ associe l'ensemble des termes clos construits au-dessus de $\kappa(\Sigma)$.*

D'après la définition précédente, pour toute signature $\Sigma \in |\mathcal{Sig}|$, la relation de satisfaction d'une Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$ dans un Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, est définie en fonction des valeurs de vérité affectées aux abstractions de formules atomiques et de l'interprétation des autres constructeurs contenus dans φ dans la $\kappa(\Sigma)$-algèbre associée à $\mathcal{M}$. Plus formellement on a:

$\models_\Sigma \subseteq \mathcal{Mod}(\Sigma) \times \mathcal{Sen}(\Sigma)$ est la relation définie de la façon suivante:
$\forall \varphi \in \mathcal{Sen}(\Sigma), \forall \mathcal{M} \in |\mathcal{Mod}(\Sigma)|,$

$$\mathcal{M} \models_\Sigma \varphi \Leftrightarrow \forall \eta \in [\![\mathcal{M}]\!]_\Sigma,\ b^{\mathcal{M}}_{\Sigma,\varphi}(\eta) = 1$$

où $b^{\mathcal{M}}_{\Sigma,\varphi} \in 2^{[\![\mathcal{M}]\!]_\Sigma}$ est définie inductivement sur la structure des formules comme suit :

- si $\varphi \in \kappa(\Sigma)^0$, alors $b^{\mathcal{M}}_{\Sigma,\varphi} = \varphi^{[\![\mathcal{M}]\!]_\Sigma}$,
- si $\varphi = c(\varphi_1, \ldots, \varphi_n)$, alors $b^{\mathcal{M}}_{\Sigma,\varphi} = c^{[\![\mathcal{M}]\!]_\Sigma}(b^{\mathcal{M}}_{\Sigma,\varphi_1}, \ldots, b^{\mathcal{M}}_{\Sigma,\varphi_n})$.

Dans la suite, nous noterons indifférement $b^{\mathcal{M}}_{\Sigma,\varphi} = 1$ ou $\mathcal{M} \models_{\Sigma,\eta} \varphi$.

Exemple 3.1.1 (Logique propositionnelle) *La κ-institution stratifiée de la logique propositionnelle est la donnée du couple $(\mathcal{SI}_{prop}, \kappa)$ tel que $\mathcal{SI}_{prop}$ est l'institution stratifiée définie dans l'exemple 1.2.1 et $\kappa : \mathcal{Sig}_{prop} \to \mathcal{LogSig}$ est le foncteur tel que pour toute signature $\Sigma \in |\mathcal{Sig}_{prop}|$, $\kappa(\Sigma) = (\mathcal{C}^n)_{n\in\mathbb{N}}$, où $(\mathcal{C}^n)_{n\in\mathbb{N}}$ est la signature de constructeurs définie par :*

- $\mathcal{C}^0 = \Sigma$;
- $\mathcal{C}^1 = \{\neg\}$;
- $\mathcal{C}^2 = \{\vee\}$;
- $\forall n > 2$, $\mathcal{C}^n = \varnothing$.

De plus, $[\![.]\!]_\Sigma : \mathcal{Mod}_{prop}(\Sigma) \to \mathcal{LogMod}(\kappa_{prop}(\Sigma))$ est un foncteur tel que pour tout Σ-modèle $v \in |2^\Sigma|$, $[\![\nu]\!]^\Sigma_{prop} = \mathcal{A}$, où $\mathcal{A} = (A, \neg^{\mathcal{A}}, \vee^{\mathcal{A}})$ est la $\kappa(\Sigma)$-algèbre définie par :

- $A = \nu$;
- $\forall p \in \Sigma$, $p^{\mathcal{A}} = \nu(p)$;
- $\forall b \in 2^\nu$, $\neg^{\mathcal{A}}(b)(\nu) = 1 - b(\nu)$;
- $\forall b, b' \in 2^\nu$, $\vee^{\mathcal{A}}(b, b')(\nu) = \max(b(\nu), b'(\nu))$.

Exemple 3.1.2 (Logique propositionnelle modale) *La κ-institution stratifiée de la logique modale propositionnelle est la donnée du couple $(\mathcal{SI}_{mod}, \kappa)$ tel que $\mathcal{SI}_{mod}$ est l'institution stratifiée définie dans l'exemple 1.2.2 et $\kappa : \mathcal{Sig}_{mod} \to \mathcal{LogSig}$ est le foncteur tel que pour toute signature $\Sigma \in |\mathcal{Sig}_{mod}|$, $\kappa(\Sigma) = (\mathcal{C}^n)_{n\in\mathbb{N}}$, où $(\mathcal{C}^n)_{n\in\mathbb{N}}$ est la signature de constructeurs définie par :*

- $\mathcal{C}^0 = \Sigma$;
- $\mathcal{C}^1 = \{\neg, \Box\}$;
- $\mathcal{C}^2 = \{\vee\}$;
- $\forall n > 2$, $\mathcal{C}^n = \varnothing$.

De plus, $[\![.]\!]_\Sigma : \mathcal{Mod}_{mod}(\Sigma) \to \mathcal{LogMod}(\kappa_{mod}(\Sigma))$ est un foncteur tel que pour tout Σ-modèle $\mathcal{M} = (\mathfrak{E}, \mathfrak{R}, \mathfrak{v}) \in |\mathcal{Mod}_{mod}(\Sigma)|$, $[\![M]\!]^\Sigma_{mod} = \mathcal{A}$, où $\mathcal{A} = (A, \neg^{\mathcal{A}}, \Box^{\mathcal{A}}, \vee^{\mathcal{A}})$ est la $\kappa(\Sigma)$-algèbre définie par :

- $A = M$;
- $\forall p \in \Sigma, \forall \eta \in \mathfrak{E}$, $p^{\mathcal{A}}(\eta) = \mathfrak{v}(p, \eta)$;
- $\forall b \in 2^M, \forall \eta \in \mathfrak{E}$, $\neg^{\mathcal{A}}(b)(\eta) = 1 - b(\eta)$;
- $\forall b \in 2^M, \forall \eta \in \mathfrak{E}$, $\Box^{\mathcal{A}}(b)(\eta) = \min\{b(\eta') / \eta \mathfrak{R} \eta'\}$;
- $\forall b, b' \in 2^M, \forall \eta \in \mathfrak{E}$, $\vee^{\mathcal{A}}(b, b')(\eta) = \max(b(\eta), b'(\eta))$.

Exemple 3.1.3 (Logique des prédicats du 1^{er}ordre) *La κ-institution stratifiée de la logique des prédicats du 1^{er}ordre est la donnée du couple $(\mathcal{SI}_{lpo}, \kappa)$ tel que $\mathcal{SI}_{lpo}$ est l'institution stratifiée définie dans l'exemple 1.2.3* *et* $\kappa : \mathcal{S}ig_{lpo} \to \mathcal{L}og\mathcal{S}ig$ *est le foncteur tel que pour toute signature* $\Sigma \in |\mathcal{S}ig_{lpo}|$, $\kappa(\Sigma) = (\mathcal{C}^n)_{n\in\mathbb{N}}$, *où* $(\mathcal{C}^n)_{n\in\mathbb{N}}$ *est la signature de constructeurs définie par :*

- $\mathcal{C}^0 = \mathcal{A}tom_\Sigma$ [1] ;
- $\mathcal{C}^1 = \{\neg\} \cup \bigcup_{x\in\mathcal{X}}\{\forall x\}$;
- $\mathcal{C}^2 = \{\vee\}$;
- $\forall n > 2, \mathcal{C}^n = \varnothing$.

De plus, $[\![.]\!]_\Sigma : \mathcal{M}od_{lpo}(\Sigma) \to \mathcal{L}og\mathcal{M}od(\kappa_{lpo}(\Sigma))$ *est un foncteur tel que pour tout* Σ*-modèle* $\mathcal{M} \in |\mathcal{M}od_{lpo}(\Sigma)|$, $[\![\mathcal{M}]\!]^\Sigma_{lpo} = \mathcal{A}$, *où* $\mathcal{A} = (A, \neg^\mathcal{A}, \vee^\mathcal{A})$ *est la* $\kappa(\Sigma)$*-algèbre définie par :*

- $A = M^\mathcal{X}$, *où* M *est l'ensemble de base de la* Σ*-algèbre* $\mathcal{M}$,
- $\forall p(t_1,\ldots,t_n) \in \mathcal{C}^0$, $p(t_1,\ldots,t_n)^\mathcal{A}(\nu) = p^\mathcal{M}(\nu^\sharp(t_1),\ldots,\nu^\sharp(t_n))$, *où* $\nu : \mathcal{X} \to M$ *est une interprétation des variables de la signature et* $\nu^\sharp : T_\Sigma(\mathcal{X}) \to M$ *son extension canonique aux* Σ*-termes,*
- $\forall \nu \in M^\mathcal{X}, \forall b \in 2^\nu$, $\neg^\mathcal{A}(b)(\nu) = 1 - b(\nu)$,
- $\forall \nu \in M^\mathcal{X}, \forall b \in 2^\nu$, $\forall x^\mathcal{A}(b)(\nu) = \mathtt{min}\{\mathtt{b}(\nu)/\forall \mathtt{y} \in \mathcal{X} \setminus \{\mathtt{x}\}, \nu'(\mathtt{y}) = \nu(\mathtt{y})\}$
- $\forall b, b' \in 2^\nu$, $\vee^\mathcal{A}(b, b')(\nu) = \max(b(\nu), b'(\nu))$.

Exemple 3.1.4 (Logique du second ordre) *La* κ*-institution stratifiée de la logique du* 2^e *ordre est la donnée du couple* $(\mathcal{SI}_{lso}, \kappa)$ *tel que* $\mathcal{SI}_{lso}$ *est l'institution stratifiée définie dans l'exemple 1.2.5 et* $\kappa : \mathcal{S}ig_{lso} \to \mathcal{L}og\mathcal{S}ig$ *est le foncteur tel que pour toute signature* $\Sigma \in |\mathcal{S}ig_{lso}|$, $\kappa(\Sigma) = (\mathcal{C}^n)_{n\in\mathbb{N}}$, *où* $(\mathcal{C}^n)_{n\in\mathbb{N}}$ *est la signature de constructeurs définie par :*

- $\mathcal{C}^0 = \mathcal{A}tom^2_\Sigma$, *où* $\mathcal{A}tom^2_\Sigma$ *est construit de la même façon que* $\mathcal{A}tom_\Sigma$ *(cf. exemple précédent) en considérant les variables de relations comme des symboles de relations, i.e.* $\mathcal{A}tom^2_\Sigma = \{r(t_1,\ldots,t_n)/r \in \mathcal{R} \cup X, t_1,\ldots,t_n \in T_\Sigma(\mathcal{X})\}$,
- $\mathcal{C}^1 = \{\neg\} \cup \bigcup_{x\in\mathcal{X}}\{\forall x\} \cup \bigcup_{\chi\in X}\{\forall \chi\}$,
- $\mathcal{C}^2 = \{\vee\}$,
- $\forall n > 2, \mathcal{C}^n = \varnothing$,

De plus, $[\![.]\!]_\Sigma : \mathcal{M}od_{lso}(\Sigma) \to \mathcal{L}og\mathcal{M}od(\kappa_{lso}(\Sigma))$ *est un foncteur tel que pour tout* Σ*-modèle* $\mathcal{M} \in |\mathcal{M}od_{lso}(\Sigma)|$, $[\![\mathcal{M}]\!]^\Sigma_{lso} = \mathcal{A}$, *où* $\mathcal{A} = (A, \neg^\mathcal{A}, \vee^\mathcal{A})$ *est la* $\kappa(\Sigma)$*-algèbre définie par :*

- $A = M^{\mathcal{X}\cup X}$, *où* M *est l'ensemble sous-jacent de la* Σ*-algèbre* $\mathcal{M}$,
- $\forall p(t_1,\ldots,t_n) \in \mathcal{C}^0$, $p(t_1,\ldots,t_n)^\mathcal{A}(\nu) = p^\mathcal{M}(\nu^\sharp(t_1),\ldots,\nu^\sharp(t_n))$ *si* $p \in \mathcal{R}$ *et* $p(t_1,\ldots,t_n)^\mathcal{A}(\nu) = \nu(p)(\nu^\sharp(t_1),\ldots,\nu^\sharp(t_n))$, *où* $\nu : \mathcal{X} \cup X \to M$ *est une interprétation des variables de la signature et* $\nu^\sharp : T_\Sigma(\mathcal{X}) \to M$ *son extension canonique aux* Σ*-termes,*
- $\forall \chi(t_1,\ldots,t_n) \in \mathcal{C}^0$, $\chi(t_1,\ldots,t_n)^\mathcal{A}(\nu) = \nu^*(\chi)^\mathcal{M}(\nu^\sharp(t_1),\ldots,\nu^\sharp(t_n))$, *où* $\nu^* = (\nu^*_n)_{n\in\mathbb{N}^*}$ *est une famille d'interprétation des variables de prédicats indexée par les naturels et telle que pour tout* $n \in \mathbb{N}^*$,
$$\begin{aligned} \nu^*_n : \ & X^n \to \wp(M^n) \\ & \chi \mapsto \nu^*_n(\chi) \subseteq M^n \end{aligned}$$

1. On rappelle que :
- $\mathcal{A}tom_\Sigma = \{p(t_1,\ldots,t_n)/p^n \in \mathcal{R} \wedge \forall 1 \le i \le n,\ t_i \in T_\Sigma(\mathcal{X})\ \}$ est l'ensemble des Σ-formules atomiques ;
- $T_\Sigma(\mathcal{X}) = \mathcal{X} \cup \{f(t_1,\ldots,t_n)/f^n \in \mathcal{F} \wedge \forall 1 \le i \le n,\ t_i \in T_\Sigma(\mathcal{X})\}$ est l'ensemble des Σ-termes.

- $\forall \nu \in M^{\mathcal{X}}, \forall b \in 2^{\nu}$, $\neg^{\mathcal{A}}(b)(\nu) = 1 - b(\nu)$,
- $\forall \nu \in M^{\mathcal{X}}, \forall b \in 2^{\nu}$, $\forall x^{\mathcal{A}}(b)(\nu) = \mathtt{min}\{\mathtt{b}(\nu)/\forall \mathtt{y} \in \mathcal{X} \setminus \{\mathtt{x}\},\ \nu'(\mathtt{y}) = \nu(\mathtt{y})\}$
- $\forall b, b' \in 2^{\nu}$, $\vee^{\mathcal{A}}(b, b')(\nu) = \max(b(\nu), b'(\nu))$.

On étend désormais la notion de morphisme d'institutions aux κ-institutions stratifiées.

Définition 3.1.6 (morphisme) *Soient deux κ-institutions stratifiées $\kappa\mathcal{SI} = (\mathcal{Sig}, \kappa\mathcal{Mod}, [\![]\!])$ et $\kappa\mathcal{SI}' = (\mathcal{Sig}', \kappa', \mathcal{Mod}', [\![]\!]')$. Un* morphisme de κ-institutions stratifiées $\mu : \mathcal{SI}' \to \mathcal{SI}$ *est la donnée d'un quadruplet $(\Phi, \alpha, \beta, \Gamma)$ tel que :*

- $\Phi : \mathcal{Sig}' \to \mathcal{Sig}$ *est un* foncteur *;*
- $\alpha : \kappa \circ \Phi \Rightarrow \kappa'$ *est une* transformation naturelle, *i.e. une famille de foncteurs indexée par $|\mathcal{Sig}'|$ telle que pour tout morphisme de signatures $\sigma' \in \mathcal{Sig}'(\Sigma'_1, \Sigma'_2)$ le diagramme de la figure 3.1 commute.*

$$
\begin{tikzcd}
\Sigma'_2 & \kappa(\Phi(\Sigma'_2)) \arrow[r, "\alpha_{\Sigma'_2}"] & \kappa'(\Sigma'_2) \\
\Sigma'_1 \arrow[u, "\sigma'"] & \kappa(\Phi(\Sigma'_1)) \arrow[u, "\kappa(\Phi(\sigma'))"] \arrow[r, "\alpha_{\Sigma'_1}"'] & \kappa'(\Sigma'_1) \arrow[u, "\kappa'(\sigma')"']
\end{tikzcd}
$$

FIG. 3.1 – *Translation de formules*

- $\beta : \mathcal{Mod}' \Rightarrow \mathcal{Mod} \circ \Phi^{op}$ *est une transformation naturelle, i.e. une famille de foncteurs indéxée par $|\mathcal{Sig}'|$ telle que pour tout morphisme de signatures $\sigma' \in \mathcal{Sig}'(\Sigma'_1, \Sigma'_2)$ le diagramme de la figure 3.2 commute.*

$$
\begin{tikzcd}
\Sigma'_2 & \mathcal{Mod}'(\Sigma'_2) \arrow[r, "\beta_{\Sigma'_2}"] \arrow[d, "\mathcal{Mod}'(\sigma')"'] & \mathcal{Mod}(\Phi(\Sigma'_2)) \arrow[d, "\mathcal{Mod}(\Phi(\sigma'))"] \\
\Sigma'_1 \arrow[u, "\sigma'"] & \mathcal{Mod}'(\Sigma'_1) \arrow[r, "\beta_{\Sigma'_1}"'] & \mathcal{Mod}(\Phi(\Sigma'_1))
\end{tikzcd}
$$

FIG. 3.2 – *Translation de modèles*

- $\Gamma = (\Gamma_{\Sigma'})_{\Sigma' \in |\mathcal{Sig}'|}$ *est une famille de transformations naturelles indexée par $|\mathcal{Sig}'|$, i.e. pour toute signature $\Sigma' \in |\mathcal{Sig}'|$, $\Gamma_{\Sigma'} : [\![]\!]'_{\Sigma'} \Rightarrow [\![]\!]_{\Phi(\Sigma')} \circ \beta_{\Sigma'}$ est une transformation naturelle et le diagramme de la figure 3.3 commute.*

On impose de plus la condition suivante :
$\forall \Sigma' \in |\mathcal{Sig}'|, \forall \varphi \in \mathcal{Sen}(\Phi(\Sigma')), \forall \mathcal{M}' \in |\mathcal{Mod}'(\Sigma')|, forall \eta' \in [\![\mathcal{M}']\!]_{\Sigma'}$,

$$\mathcal{M}' \models'_{\Sigma', \eta'} \alpha^{\sharp}_{\Sigma'}(\varphi) \Leftrightarrow \beta_{\Sigma'}(\mathcal{M}') \models_{\Phi(\Sigma'), \Gamma_{\Sigma', \mathcal{M}'}(\eta)} \varphi$$

où $\alpha^{\sharp}_{\Sigma'}$ est l'extension du foncteur $\alpha_{\Sigma'}$ à l'ensemble $T_{\kappa'(\Sigma')}$ des termes construits sur la signature de constructeurs $\kappa'(\Sigma')$, i.e. les Σ'-formules.

$$\begin{tikzcd}
\mathcal{M}'_2 & \llbracket \mathcal{M}'_2 \rrbracket'_{\Sigma'} \arrow[r, "\Gamma_{\Sigma',\mathcal{M}'_2}"] \arrow[d, "\llbracket \mu \rrbracket"'] & \llbracket \beta_{\Sigma'}(\mathcal{M}'_2) \rrbracket_{\Phi(\Sigma')} \arrow[d, "\llbracket \beta_{\Sigma'}(\mu) \rrbracket"] \\
\mathcal{M}'_1 \arrow[u, "\mu'"] & \llbracket \mathcal{M}'_1 \rrbracket_{\Sigma'} \arrow[r, "\Gamma_{\Sigma',\mathcal{M}'_1}"'] & \llbracket \beta_{\Sigma'}(\mathcal{M}'_1) \rrbracket_{\Phi(\Sigma')}
\end{tikzcd}$$

FIG. 3.3 – *Translation de modèles*

Proposition 3.1.1 *Les morphismes de κ-institutions stratifiées présentent les morphismes d'institutions de la même façon que les κ-institutions stratifiées présentent les institutions, i.e. si $\mu : \kappa\mathcal{SI}' \to \kappa\mathcal{SI}$ est un morphisme de κ-institutions stratifiées et si $\kappa\mathcal{SI}$ et $\kappa\mathcal{SI}'$ présentent respectivement les institutions $\mathcal{I}$ et $\mathcal{I}'$, alors il existe un morphisme d'institution $\mu' : \mathcal{I}' \to \mathcal{I}$.*

3.2 Théorème des chaînes élémentaires

Étant donné un morphisme $\mu : \mathcal{M} \to \mathcal{M}'$ de Σ-modèles, le morphisme $\llbracket \mu \rrbracket_\Sigma : \llbracket \mathcal{M} \rrbracket_\Sigma \to \llbracket \mathcal{M}' \rrbracket_\Sigma$ n'est pas nécessairement un morphisme de $\kappa(\Sigma)$-algèbres, sauf dans le cas où le morphisme μ est élémentaire. Dans ce cas, on a le résultat suivant.

Proposition 3.2.1 *Soit $\kappa\mathcal{SI} = (\mathcal{S}ig, \mathcal{M}od, \llbracket\rrbracket, \models, \kappa)$ une κ-institution stratifiée. Pour toute signature $\Sigma \in |\mathcal{S}ig|$, le morphisme $\llbracket \mu \rrbracket_\Sigma : \llbracket \mathcal{M} \rrbracket_\Sigma \to \llbracket \mathcal{M}' \rrbracket_\Sigma$ est un $\kappa(\Sigma)$-morphisme si et seulement si $\mu : \mathcal{M} \to \mathcal{M}'$ est un morphisme de $\mathcal{E}lem(\Sigma)$, i.e. :*
$\forall \varphi \in \mathcal{S}en(\Sigma), \forall \eta \in \llbracket \mathcal{M} \rrbracket_\Sigma,$

$$\mathcal{M} \models_{\Sigma,\eta} \varphi \iff \mathcal{M}' \models_{\Sigma,\llbracket \mu \rrbracket_\Sigma(\eta)} \varphi$$

Preuve Pour toute signature $\Sigma \in |\mathcal{S}ig|$, tout Σ-modèle $\mathcal{M} \in |Mod(\Sigma)|$ et toute Σ-formule $\varphi \in \mathcal{S}en(\Sigma)$, soit $b^{\mathcal{M}}_{\Sigma,\varphi} \in 2^{\llbracket \mathcal{M} \rrbracket_\Sigma}$ l'application définie de la façon suivante :

$$\mathcal{M} \models_{\Sigma,\eta} \varphi \iff b^{\mathcal{M}}_{\Sigma,\varphi}(\eta) = 1$$

La preuve se fait alors par induction sur la structure des formules.
cas de base : Soit $\varphi \in C_0$. Par la définition 3.1.3, on a :

$$\forall \eta \in \llbracket \mathcal{M} \rrbracket_\Sigma,\ \varphi^{\llbracket \mathcal{M} \rrbracket_\Sigma}(\eta) = \varphi^{\llbracket \mathcal{M}' \rrbracket_\Sigma}(\llbracket \mu \rrbracket_\Sigma(\eta)) \iff b^{\mathcal{M}}_{\Sigma,\varphi}(\eta) = b^{\mathcal{M}'}_{\Sigma,\varphi}(\llbracket \mu \rrbracket_\Sigma(\eta))$$

cas général : Soit φ une Σ-formule de la forme $c(\varphi_1, \ldots, \varphi_n)$. Par hypothèse d'induction, pour tout $1 \leq i \leq n$ on a :

$$\forall \eta \in \llbracket \mathcal{M} \rrbracket_\Sigma,\ b^{\mathcal{M}}_{\Sigma,\varphi_i}(\eta) = b^{\mathcal{M}'}_{\Sigma,\varphi_i}(\llbracket \mu \rrbracket_\Sigma(\eta))$$

Alors, par la définition 3.1.3, on a :
$\forall \eta \in \llbracket \mathcal{M} \rrbracket_\Sigma,$

$$c^{\llbracket \mathcal{M} \rrbracket_\Sigma}(b^{\mathcal{M}}_{\Sigma,\varphi_1}, \ldots, b^{\mathcal{M}}_{\Sigma,\varphi_n})(\eta) = c^{\llbracket \mathcal{M}' \rrbracket_\Sigma}(b^{\mathcal{M}'}_{\Sigma,\varphi_1}{}_{|_{\llbracket \mathcal{M} \rrbracket_\Sigma}}, \ldots, b^{\mathcal{M}'}_{\Sigma,\varphi_n}{}_{|_{\llbracket \mathcal{M} \rrbracket_\Sigma}})(\llbracket \mu \rrbracket_\Sigma(\eta))$$
$$\Updownarrow$$
$$b^{\mathcal{M}}_{\Sigma,\varphi}(\eta) = b^{\mathcal{M}'}_{\Sigma,\varphi}(\llbracket \mu \rrbracket_\Sigma(\eta))$$

□

Par la proposition 3.2.1, le morphisme $[\![]\!]_\Sigma$ défini alors un foncteur de $\mathcal{E}lem(\Sigma)$ dans $\mathcal{L}og\mathcal{M}od(\kappa(\Sigma))$.

Définition 3.2.1 (chaîne) *Une* chaîne *de $\mathcal{C}$-algèbres est une séquence croissante de $\mathcal{C}$-algèbres dans $\mathcal{L}og\mathcal{M}od(\mathcal{C})$*

$$\mathcal{A}_0 \xrightarrow{\mu_1} \mathcal{A}_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_\beta} \mathcal{A}_\beta \xrightarrow{\mu_{\beta+1}} \ldots, \beta < \alpha$$

dont la longueur est un ordinal α.

Définition 3.2.2 (union de chaîne) *Soit $\mathcal{C} = \mathcal{A}_0 \xrightarrow{\mu_1} \mathcal{A}_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_\beta} \mathcal{A}_\beta \xrightarrow{\mu_{\beta+1}} \ldots, \beta < \alpha$ une chaîne de $\mathcal{C}$-algèbres. L'*union de la chaîne $\mathcal{C}$ (modulo $\mu_1, \ldots, \mu_\beta, \ldots$) *est la $\mathcal{C}$-algèbre $\mathcal{A} = \bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$ dont l'ensemble de base est l'ensemble $A = \bigcup_{\beta<\alpha} A_\beta/_{\equiv_\alpha}$ où $\equiv_\alpha$ est la relation d'équivalence définie sur $\bigcup_{\beta<\alpha} A_\beta$ de la façon suivante :*
$\forall \eta \in A_\beta, \forall \eta' \in A_\gamma$

$$\eta \equiv_\alpha \eta' \Longleftrightarrow \exists \delta,\ \gamma, \beta < \delta, \mu_\delta(\ldots(\mu_{\beta+1}(\eta))\ldots) = \mu_\delta(\ldots(\mu_{\gamma+1}(\eta'))\ldots)$$

On remarquera que la relation $\equiv_\alpha$ est clairement une relation d'équivalence. Soit $[\eta]_{\equiv_\alpha} \in A$. Par définition ,il existe $\beta < \alpha$ tel que $\eta \in A_\beta$. On pose alors pour tout $c^n \in \mathcal{C}$:

$$c^{\mathcal{A}}(b_1, \ldots, b_n)([\eta]_{\equiv_\alpha}) = c^{\mathcal{A}}_\beta(b_{1|_{\mathcal{A}_\beta}}, \ldots, b_{n|_{\mathcal{A}_\beta}})(\eta)$$

où $b_{i|_{\mathcal{A}_\beta}} : \eta \mapsto b_i([\eta]_{\equiv_\alpha})$.
La décision ne dépend pas du choix de η (et donc de l'indice β). En effet, si η' est tel que $\eta \equiv_\alpha \eta'$ et $\eta' \in A_\gamma$ pour un certain $\gamma < \alpha$, alors il existe $\beta, \gamma < \delta$ tel que $\mu_\delta(\ldots(\mu_{\beta+1}(\eta))\ldots) = \mu_\delta(\ldots(\mu_{\gamma+1}(\eta))\ldots)$. Chacun des $\mu_1, \ldots, \mu_\beta, \ldots$ étant des $\mathcal{C}$-morphismes, dans les deux cas, on a :

$$c^{\mathcal{A}_\beta}(b_{1|_{\mathcal{A}_\beta}}, \ldots, b_{n|_{\mathcal{A}_\beta}})(\eta) = c^{\mathcal{A}_\gamma}(b_{1|_{\mathcal{A}_\gamma}}, \ldots, b_{n|_{\mathcal{A}_\gamma}})(\eta')$$

Lemme 3.2.1 *Soit $J : \alpha \to \mathcal{L}og\mathcal{M}od(\mathcal{C})$ un foncteur où α est un ordinal. Alors, $\bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$ est la colimite pour J dans $\mathcal{L}og\mathcal{M}od(\mathcal{C})$.*

Preuve Soit $\bigcup_{\beta<\alpha} \mathcal{A}_\beta$ la $\mathcal{C}$-algèbre dont l'ensemble de base est $\bigcup_{\beta<\alpha} A_\beta$ et telle que pour tout constructeur $c^n \in \mathcal{C}$ et tout $(b_1, \ldots, b_n) \in (2^{\bigcup_{\beta<\alpha} A_\beta})^n$, $c^{\bigcup_{\beta<\alpha} \mathcal{A}_\beta}(b_1, \ldots, b_n) = \bigcup_{\beta<\alpha} c^{\mathcal{A}_\beta}(b_{1|_{\mathcal{A}_\beta}}, \ldots, b_{n|_{\mathcal{A}_\beta}})$. Par construction, $\bigcup_{\beta<\alpha} \mathcal{A}_\beta$ est le plus petit $\mathcal{C}$-modèle contenant chaque $\mathcal{A}_\beta$ comme $\mathcal{C}$-sous-modèle dans

$\mathcal{LogMod}(\mathcal{C})$, *i.e.* il existe un $\mathcal{C}$-morphisme inclusif dans $\mathcal{LogMod}(\mathcal{C})$, de $\mathcal{A}_\beta$ dans $\bigcup_{\beta<\alpha} \mathcal{A}_\beta$. Par construction de $\bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$, l'application $q : \bigcup_{\alpha<\beta} \mathcal{A}_\beta \to \bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$ qui associe à tout $\eta \in \bigcup_{\alpha<\beta} \mathcal{A}_\beta$ sa classe d'équivalence $[\eta]_{\equiv_\alpha} \in \bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$ est un $\mathcal{C}$-morphisme dans $\mathcal{LogMod}(\mathcal{C})$. Soit alors $q_\beta = q_{|\mathcal{A}_\beta}$. Montrer que q_β est un $\mathcal{C}$-morphisme dans $\mathcal{LogMod}(\mathcal{C})$ est alors évident. Enfin, par la propriété d'universalité associée aux morphismes quotients, $\bigcup_{\beta<\alpha} \mathcal{A}_\beta/_{\equiv_\alpha}$ est la colimite ordinale pour J dans $\mathcal{LogMod}(\mathcal{C})$. □

Pour obtenir le théorème des chaînes élémentaires, nous avons besoin de la condition supplémentaire suivante.

Définition 3.2.3 (propriété d'union de chaine) *Soit* $\mathcal{SI} = (\mathcal{Sig}, \mathcal{Mod}, [\![]\!], \models, \kappa)$ *une κ-institution stratifiée qui possède des colimites ordinales de modèles. $\mathcal{SI}$ possède la* propriété d'union de chaîne *si pour toute signature $\Sigma \in |\mathcal{Sig}|$ et tout foncteur $J : \alpha \to \mathcal{Mod}(\Sigma)$, où α est un ordinal, $[\![]\!]_\Sigma$ préserve les colimites ordinales pour J dans $\mathcal{LogMod}(\kappa(\Sigma))$ et $\mathcal{Mod}(\Sigma)$.*

la colimite $(\mathcal{M}, \mu)$ de J dans $\mathcal{Mod}(\Sigma)$ vérifie que $[\![\mathcal{M}]\!]_\Sigma$ et $\bigcup_{\beta<\alpha} [\![J(\beta)]\!]_\Sigma/_{\equiv_\alpha}$ sont isomorphes .

La plupart des logiques vérifie la propriété de l'union des chaînes. Par exemple :

- en logique propositionnelle c'est évident car les morphismes sont des identités ;
- en logique propositionnelle modale, étant donnée une séquence croissante

$$(\mathfrak{E}_0, \mathfrak{R}_0, \mathfrak{v}_0) \xrightarrow{\mu_1} (\mathfrak{E}_1, \mathfrak{R}_1, \mathfrak{v}_1) \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_\beta} (\mathfrak{E}_\beta, \mathfrak{R}_\beta, \mathfrak{v}_\beta) \xrightarrow{\mu_{\beta+1}} \ldots, \ \beta < \alpha$$

  la colimite ordinale dans $\mathcal{Mod}(\Sigma)$ est la structure de Kripke $(\mathfrak{E}, \mathfrak{R}, \mathfrak{v})$ définie comme suit :
  - $\mathfrak{E} = \bigcup_{\beta<\alpha} \mathfrak{E}_\beta/_{\equiv_\alpha}$;
  - $\mathfrak{R} = \bigcup_{\beta<\alpha} \mathfrak{R}_\beta/_{\equiv_\alpha}$, où $[\eta]_{\equiv_\alpha}\mathfrak{R}[\eta']_{\equiv_\alpha} \Leftrightarrow (\exists \beta < \alpha, \exists \eta_1 \in [\eta]_{\equiv_\alpha} \cap \mathfrak{E}_\beta, \exists \eta_2 \in [\eta']_{\equiv_\alpha} \cap \mathfrak{E}_\beta), \eta_1 \mathfrak{R}_\beta \eta_2)$;
  - $\mathfrak{v} : \mathfrak{E} \times \Sigma \to \{0,1\}$ associe $([\eta]_{\equiv_\alpha}, p)$ à $\mathfrak{v}_\beta(\eta, p)$ pour tout $\eta \in \mathfrak{E}_\beta$, où $\beta < \alpha$.

  Puisque pour tout $\beta < \alpha$ μ_β Σ-morphisme, *i.e.* il préserve à la fois les relations d'accessiblité et les valeurs de vérité, l'application correspondante $q_\beta : S_\beta \to S$ qui, à tout s, associe $[s]_{\equiv_\Sigma}$ est un Σ-morphisme. Par construction, (S, R, ν) vérifie : $[\![(S, R, \nu)]\!]_\Sigma = \bigcup_{\beta<\alpha} [\![(S_\beta, R_\beta, \nu_\beta)]\!]_\Sigma/_{\equiv_\alpha}$.
- En logique du premier ordre, étant donnée une séquence croissante

$$\mathcal{M}_0 \xrightarrow{\mu_1} \mathcal{M}_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_\beta} \mathcal{M}_\beta \xrightarrow{\mu_{\beta+1}} \ldots, \ \beta < \alpha$$

la limite ordinale dans $\mathcal{M}od(\Sigma)$ est la Σ-structure $\mathcal{M}$ définie de la façon suivante :

- $\mathcal{M} = \bigcup_{\beta<\alpha} M_\beta/_{\equiv'_\alpha}$ où, $\equiv'_\alpha$ est la relation d'équivalence définie sur $\bigcup_{\beta<\alpha} M_\beta$ comme suit[2]:
  $\forall m \in M_\beta, \forall m' \in A_\gamma,$
  $$m \equiv'_\alpha m' \iff \exists \delta, \gamma,\ \beta < \delta,\ \mu_\delta(\ldots(\mu_{\beta+1}(m))\ldots) = \mu_\delta(\ldots(\mu_{\gamma+1}(m'))\ldots)$$
- $\forall f^n \in F, f^{\mathcal{M}} = \bigcup_{\beta<\alpha} f_\beta^{\mathcal{M}}/_{\equiv'_\alpha}$
- $\forall p^n \in P, p^{\mathcal{M}} = \bigcup_{\beta<\alpha} p^{\mathcal{M}_\beta}/_{\equiv'_\alpha}$.

Puisque pour tout $\beta < \alpha$, μ_β est un Σ-morphisme, *i.e.* il préserve à la fois les interprétations des symboles de fonction et de prédicat l'application correspondante $q_\beta : \mathcal{M}_\beta \to \mathcal{M}$ qui, à tout m associe $[m]_{\equiv'_\alpha}$ est un Σ-morphisme.
Soit $\approx_\alpha$ la relation d'équivalence définie sur $\bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma$ de la façon suivante :
$$\iota \approx_\alpha \iota' \iff (\forall x \in V,\ \iota(x) \equiv'_\alpha \iota'(x))$$
Il est clair que : $\iota \approx_\alpha \iota' \iff \iota \equiv_\alpha \iota'$.
D'où, on a $\bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma/_{\approx_\alpha}$ et $\bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma/_{\equiv_\alpha}$ sont isomporphes. On considère alors l'application $\delta : [\![\mathcal{M}]\!]_\Sigma \to \bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma/_{\approx_\alpha}$ qui associe à tout $\iota \in M^V$, $[\iota']_{\approx_\alpha}$, où pour tout $x \in V$, $\iota(x) = [\iota'(x)]_{\equiv'_\alpha}$. Ceci définit effectivement une application. En effet, si pour tout $x \in V$, $\iota(x) = [\iota'(x)]_{\equiv'_\alpha} = [\iota''(x)]_{\equiv'_\alpha}$ alors $\iota' \approx_\alpha \iota''$. De plus, δ est une bijection. Supposons que pour tous $\iota_1, \iota_2 \in M^V$, $\delta(\iota_1) = \delta(\iota_2)$. Par définition, $\delta(\iota_1) = [\iota'_1]_{\approx_\alpha}$, $\delta(\iota_2) = [\iota'_2]_{\approx_\alpha}$ et donc pour tout $x \in V$, $[\iota'_1(x)]_{\equiv'_\alpha} = [\iota'_2(x)]_{\equiv'_\alpha}$, *i.e.* $\iota_1(x) = \iota_2(x)$.
Par définition de $\approx_\alpha$, δ est surjective.
D'où $[\![\mathcal{M}]\!]_\Sigma$ et $\bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma /_{\equiv_\alpha}$ sont en bijection. Par une induction simple sur les formules on montre de plus que δ est un isomorphisme dans $\mathcal{L}og\mathcal{M}od(C(\Sigma))$.

- En logique observationnelle, étant donnée une séquence croissante
$$\mathcal{M}_0 \xrightarrow{\mu_1} \mathcal{M}_1 \xrightarrow{\mu_2} \ldots \xrightarrow{\mu_\beta} \mathcal{M}_\beta \xrightarrow{\mu_{\beta+1}} \ldots,\ \beta < \alpha$$
la limite ordinale $\mathcal{M}$ est définie de la même façon qu'en logique du premier ordre sauf que les prédicats sont oubliés. De plus, parceque les ensembles de contextes $\mathcal{C}t\chi$ sont similaires dans tous les $[\![\mathcal{M}_\beta]\!]_\Sigma$ $(\beta < \alpha)$, en suivant la même démarche que dans l'exemple précédent, on montre que $[\![\mathcal{M}]\!]_\Sigma$ et $\bigcup_{\beta<\alpha} [\![\mathcal{M}_\beta]\!]_\Sigma/_{\equiv_\alpha}$ sont isomorphes dans $\mathcal{L}og\mathcal{M}od(C(\Sigma))$.

2. Il ne faut pas confondre $\equiv'_\alpha$ avec $\equiv_\alpha$ qui est définie sur les interprétations des variables dans les Σ-structures.

Théorème 3.2.1 (théorème des chaînes élémentaires) *Soit $\mathcal{SI}$ une institution stratifiée possédant la propriété de l'union de chaînes. Soit $\Sigma \in |\mathcal{Sig}|$ une signature. Alors, $\mathcal{Elem}(\Sigma)$ est fermée par colimite ordinale.*

Preuve Soit α un ordinal et soit $J : \alpha \to \mathcal{Elem}(\Sigma)$ un foncteur. Par hypothèse, J possède une colimite $(\mathcal{M}, \mu)$ dans $\mathcal{Mod}(\Sigma)$. Puisque $\mathcal{SI}$ possède la propriété d'union de chaines, $[\![\mathcal{M}]\!]_\Sigma$ est isomorphe à $\bigcup\limits_{\beta<\alpha} [\![J(\beta)]\!]_\Sigma/_{\equiv_\alpha}$ dans $\mathcal{LogMod}(\kappa(\Sigma))$. Alors, par le lemme 3.2.1, $[\![\mu_\beta]\!]_\Sigma : [\![J(\beta)]\!]_\Sigma \to [\![\mathcal{M}]\!]_\Sigma$ est un $\mathcal{C}$-morphisme dans $\mathcal{LogMod}(\kappa(\Sigma))$. Par la proposition 3.2.1, μ_β est alors élémentaire. □

3.3 Théorème de consistance de Robinson

Nous arrivons maintenant au résultat principale de cette partie : une preuve institution-indépendante de la consistance de Robinson.
On suppose dans la suite qu'une institution stratifiée $\mathcal{SI} = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, [\![]\!])$ vérifie les conditions suivantes :

1. $\mathcal{SI}$ est compacte et possède des diagrammes complets, la propriété des chaînes élémentaires, la négation et la conjonction finie[3]
2. $\mathcal{SI}$ est fermée par isomorphisme, *i.e.* pour toute signature $\Sigma \in |\mathcal{Sig}|$ et tous Σ-modèles isomorphes $\mathcal{M}$ et $\mathcal{M}'$, $\mathcal{M} \equiv_\Sigma \mathcal{M}'$.
3. $\mathcal{Sig}$ possède des pushouts.

Théorème 3.3.1 (théorème de consistance de Robinson) *Soit*

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle\sigma_2}\downarrow & & \downarrow{\scriptstyle\sigma'_1} \\ \Sigma_2 & \xrightarrow[\sigma'_2]{} & \Sigma' \end{array}$$

un diagramme commutatif d'amalgation faible de morphismes de signatures élevant les isomorphismes. Soit T une Σ-théorie complète et soient $T_i \supseteq \mathcal{Sen}(\sigma_i)(T)$ $(i = 1, 2)$ des Σ_i-théories consistantes. Alors, $T' = \mathcal{Sen}(\sigma'_1)(T_1) \cup \mathcal{Sen}(\sigma'_2)(T_2)$ est une Σ'-théorie consistante.

Preuve La preuve de ce théorème est une généralisation de la preuve établie dans le cadre de la théorie standard des modèles. Elle repose sur les trois lemmes suivants.

Lemme 3.3.1 *Soit $\mathcal{M} \in \mathcal{Mod}(T)$. Alors, il existe un modèle $\mathcal{M}'' \in \mathcal{Mod}(T_2)$ et un Σ-morphisme $\mu : \mathcal{M} \to \mathcal{Mod}(\sigma_2)(\mathcal{M}'')$ tels que $\mathcal{M} \prec_\mu \mathcal{Mod}(\sigma_2)(\mathcal{M}'')$.*

Preuve Soit $\mathcal{D}(\mathcal{M})$ le diagramme complet de $\mathcal{M}$. Puisque $\mathcal{Sig}$ possède des pushouts, le pushout de signatures suivant existe.

$$\begin{array}{ccc} \Sigma & \xrightarrow{\iota_\Sigma(\mathcal{M})} & \Sigma_\mathcal{M} \\ {\scriptstyle\sigma_2}\downarrow & & \downarrow{\scriptstyle\rho_1} \\ \Sigma_2 & \xrightarrow[\rho_2]{} & \Sigma' \end{array}$$

3. Les hypothèses de compacité et de conjonction finie peuvent être indifféremment remplacées par celle de conjonction infinie.

Montrons alors que $\mathit{Sen}(\rho_2)(T_2) \cup \mathit{Sen}(\rho_1)(D(\mathcal{M}))$ et consistant. Si on suppose le contraire, ceci signife, par compacité, conjonction finie et le fait que $\mathcal{D}(\mathcal{M})$ est fermé par conjonction fini, qu'il existe $\alpha(\varphi,\eta) \in \mathcal{D}(\mathcal{M})$ tel que $\mathit{Sen}(\rho_2)(T_2) \models_{\Sigma'} \neg\mathit{Sen}(\rho_1)(\alpha(\varphi,\eta))$. Soit $\mathcal{M}' \in \mathit{Mod}(\mathit{Sen}(\rho_2)(T_2))$. Par la condition de satisfaction, $\mathit{Mod}(\rho_1)(\mathcal{M}') \models_{\Sigma_\mathcal{M}} \neg\alpha(\varphi,\eta)$. Alors, par le deuxiÃÂ¨me point de la définition 2.2.2, on a :

$$\mathit{Mod}(\rho_1 \circ \iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{\Sigma,[\![i_{\Sigma,\mathcal{M}}(\mathit{Mod}(\rho_1)(\mathcal{M}'))]\!]_\Sigma(\eta)} \neg\varphi$$

Puisque T est complète et que $\mathit{Sen}(\sigma_2)(T) \subseteq T_2$, $\mathit{Mod}(\rho_1 \circ \iota_\Sigma(\mathcal{M}))(\mathcal{M}') \in \mathit{Mod}(T)$ et donc $\neg\varphi \in T$. Seulement, $\mathcal{M} \models_{\Sigma,\eta} \varphi$, et donc, puisque T est complète et que $\mathcal{M} \in \mathit{Mod}(T)$, $\varphi \in T$ ce qui est une contradiction.
Il existe alors $\mathcal{M}' \in \mathit{Mod}(\mathit{Sen}(\rho_2)(T_2) \cup \mathit{Sen}(\rho_1)(D(\mathcal{M})))$. Soit $\mathcal{M}'' = \mathit{Mod}(\rho_2)(\mathcal{M}')$. Puisque $\mathit{Mod}(\rho_1)(\mathcal{M}') \in \mathit{Mod}(D(\mathcal{M}))$ et $\mathit{Mod}(\sigma_2 \circ \rho_2)(\mathcal{M}') = \mathit{Mod}(\iota_\Sigma(\mathcal{M}) \circ \rho_1)(\mathcal{M}')$, par le théorème 2.2.1, on a $\mathcal{M} \prec_{i_{\Sigma,\mathcal{M}}(\mathit{Mod}(\rho_1)(\mathcal{M}'))} \mathit{Mod}(\sigma_2)(\mathcal{M}'')$. □

Lemme 3.3.2 *Soit $\mathcal{M}_1 \in \mathit{Mod}(T_1)$, $\mathcal{M} \in \mathit{Mod}(T)$ et supposons que $\mathit{Mod}(\sigma_1)(\mathcal{M}_1) \prec_{\mu_1} \mathcal{M}$. Alors, il existe $\mathcal{M}_2 \in \mathit{Mod}(\Sigma_1)$ tel que $\mathcal{M}_1 \prec_\mu \mathcal{M}_2$ et $\mathcal{M} \prec_{\mu_2} \mathit{Mod}(\sigma_1)(\mathcal{M}_2)$ (et donc $\mathcal{M}_2 \in \mathit{Mod}(T_1)$).*

Preuve Puisque Sig possède des pushouts, le pushout de signatures suivant existe.

$$\begin{tikzcd}
\Sigma \arrow[r, "\iota_\Sigma(\mathcal{M})"] \arrow[d, "\iota_{\Sigma_1}(\mathcal{M}_1)\circ\sigma_1"'] & \Sigma_\mathcal{M} \arrow[d, "\rho_1"] \\
\Sigma_{\mathcal{M}_1} \arrow[r, "\rho_2"'] & \Sigma'
\end{tikzcd}$$

Montrons alors que $\mathit{Sen}(\rho_1)(D(\mathcal{M})) \cup \mathit{Sen}(\rho_2)(\mathcal{D}(\mathcal{M}_1))$ est consistant. Supposons le contraire. Il existe alors $\alpha(\varphi,\eta) \in \mathcal{D}(\mathcal{M})$ tel que $\mathit{Sen}(\rho_2)(\mathcal{D}(\mathcal{M}_1)) \models_{\Sigma'} \neg\mathit{Sen}(\rho_1)(\alpha(\varphi,\eta))$. Soit $\mathcal{M}' \in \mathit{Mod}(\mathit{Sen}(\rho_2)(\mathcal{D}(\mathcal{M}_1)))$. Par la condition de satisfaction, on a $\mathit{Mod}(\rho_1)(\mathcal{M}') \models_{\Sigma_\mathcal{M}} \neg\alpha(\varphi,\eta)$. Alors, par le second point de la définition 2.2.2, on a :

$$(1) \quad \mathit{Mod}(\rho_1 \circ \iota_\Sigma(\mathcal{M}))(\mathcal{M}') \models_{\Sigma,[\![i_{\Sigma,\mathcal{M}}(\mathit{Mod}(\rho_1)(\mathcal{M}'))]\!]_\Sigma(\eta)} \neg\varphi$$

Par hypothèse et par la condition de satisfaction, on a également $\mathit{Mod}(\rho_2)(\mathcal{M}') \in \mathcal{D}(\mathcal{M}_1)$. Alors, par le théorème 2.2.1, on a :

$$\mathcal{M}_1 \prec_{i_{\Sigma_1,\mathcal{M}_1}(\mathit{Mod}(\rho_2)(\mathcal{M}'))} \mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1))(\mathcal{M}')$$

er donc $\mathcal{M}_1 \equiv_{\Sigma_1} \mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1))(\mathcal{M}')$.
On a alors $\mathit{Mod}(\sigma_1)(\mathcal{M}_1) \equiv_\Sigma \mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1) \circ \sigma_1)(\mathcal{M}')$, et donc $\mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1) \circ \sigma_1)(\mathcal{M}') \in \mathit{Mod}(T)$. Puisque T est complète, on a $\mathcal{M} \equiv_\Sigma \mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1) \circ \sigma_1)(\mathcal{M}')$.
Par (1), on en conclut alors $\mathcal{M} \models_\Sigma \neg\varphi$, ce qui contredit $\alpha(\varphi,\eta) \in D(\mathcal{M})$. Il existe alors $\mathcal{M}' \in \mathit{Mod}(\mathit{Sen}(\rho_1)(\mathcal{D}(\mathcal{M})) \cup \mathit{Sen}(\rho_2)(\mathcal{D}(\mathcal{M}_1)))$. Puisque $\mathcal{SI}$ présente une institution, $\mathit{Mod}(\rho_1)(\mathcal{M}') \in \mathit{Mod}(\mathcal{D}(\mathcal{M}))$ et $\mathit{Mod}(\rho_2)(\mathcal{M}') \in \mathit{Mod}(\mathcal{D}(\mathcal{M}_1))$. Soit $\mathcal{M}_2 = \mathit{Mod}(\rho_2 \circ \iota_{\Sigma_1}(\mathcal{M}_1))(\mathcal{M}')$. Par le théorème 2.2.1, on a alors $\mathcal{M}_1 \prec_{i_{\Sigma_1,\mathcal{M}_1}(\mathit{Mod}(\rho_2)(\mathcal{M}'))} \mathcal{M}_2$ et $\mathcal{M} \prec_{i_{\Sigma,\mathcal{M}}(\mathit{Mod}(\rho_1)(\mathcal{M}'))} \mathit{Mod}(\sigma_1)(\mathcal{M}_2)$. □

Bien entendu, on peut remplacer T_1 par T_2 :

Lemme 3.3.3 *Soient $\mathcal{N}_1 \in \mathcal{Mod}(T_2)$, $\mathcal{M} \in \mathcal{Mod}(T)$ et supposons que $\mathcal{Mod}(\sigma_2)(\mathcal{N}_1) \prec_{\mu_1} \mathcal{M}$. Alors, il existe $\mathcal{N}_2 \in \mathcal{Mod}(\Sigma_1)$ tel que $\mathcal{N}_1 \prec_{\mu} \mathcal{N}_2$ et $\mathcal{M} \prec_{\mu_2} \mathcal{Mod}(\sigma_1)(\mathcal{N}_2)$ (et donc $\mathcal{N}_2 \in \mathcal{Mod}(T_2)$).*

On peut maintenant finir la preuve du théorème en construisant un Σ'-modèle de $\mathcal{Sen}(\sigma_1')(T_1) \cup \mathcal{Sen}(\sigma_2')(T_2)$. On commence par un Σ_1-modèle $\mathcal{M}_1 \in \mathcal{Mod}(T_1)$. Puisque $\mathcal{SI}$ présente une institution, $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \in \mathcal{Mod}(T)$. Par le lemme 3.3.1, il existe $\mathcal{N}_1 \in \mathcal{Mod}(T_2)$ et $\mu_1 : \mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \to \mathcal{Mod}(\sigma_2)(\mathcal{N}_1)$ tel que $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \prec_{\mu_1} \mathcal{Mod}(\sigma_2)(\mathcal{N}_1)$. On applique alors le lemme 3.3.2 et on trouve un Σ_1-modèle $\mathcal{M}_2 \in \mathcal{Mod}(T_1)$ tel que $\mathcal{M}_1 \prec_{\nu_2} \mathcal{M}_2$ et $\mathcal{Mod}(\sigma_2)(\mathcal{N}_1) \prec_{\rho_2} \mathcal{Mod}(\sigma_1)(\mathcal{M}_2)$. On applique alors alternativement les lemmes 3.3.3 et 3.3.2 et on trouve des Σ_1-modèles $\mathcal{M}_n \in \mathcal{Mod}(T_1)$ et des Σ_2-modèles $\mathcal{N}_n \in \mathcal{Mod}(T_2)$ tels que pour tout $n \in \mathbb{N}$:

$$\mathcal{M}_n \prec_{\mu_{n+1}} \mathcal{M}_{n+1}, \quad \mathcal{N}_n \prec_{\nu_{n+1}} \mathcal{N}_{n+1}, \text{ et}$$
$$\mathcal{Mod}(\sigma_1)(\mathcal{M}_n) \prec_{\tau_n} \mathcal{Mod}(\sigma_2)(\mathcal{N}_n), \quad \mathcal{Mod}(\sigma_2)(\mathcal{N}_n) \prec_{\rho_{n+1}} \mathcal{Mod}(\sigma_1)(\mathcal{M})_{n+1})$$

Soit $J_i : \omega \to \mathcal{Elem}(\Sigma_i)$ $i = 1, 2$ le foncteur ordinal qui associe $\mathcal{M}_n$ (resp. $\mathcal{N}_n$) à n et μ_{i+1} (resp. ν_{i+1}) à $i < i+1$. Puisque $\mathcal{SI}$ possède la propriété des chaines élémentaires, J_1 et J_2 ont une limite ordinale $\mathcal{M}$ et $\mathcal{N}$, respectivement, et alors $\mathcal{M} \in \mathcal{Mod}(T_1)$ et $\mathcal{N} \in \mathcal{Mod}(T_2)$. Maintenant, on a également la séquence croissante suivante dans $\mathcal{Elem}(\Sigma)$:

$$\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) \xrightarrow{\mu_1} \mathcal{Mod}(\sigma_2)(\mathcal{N}_2) \xrightarrow{\rho_2} \mathcal{Mod}(\sigma_1)(\mathcal{M}_2) \xrightarrow{\mu_2} \ldots$$

Puisque les foncteurs respectent les colimites, $\mathcal{Mod}(\sigma_1)(\mathcal{M})$ et $\mathcal{Mod}(\sigma_2)(\mathcal{N})$ sont toutes deux des colimites ordinales de la séquence ci-dessus. Ils sont donc isomorphes. De plus, puisque le diagramme ci-dessus élève les isomorphismes, il existe $\mathcal{M}' \in \mathcal{Mod}(\Sigma_1)$ et $\mathcal{N}' \in \mathcal{Mod}(\Sigma_2)$, respectivement isomorphes à $\mathcal{M}$ et $\mathcal{N}$ et tels que $\mathcal{Mod}(\sigma_1)(\mathcal{M}') = \mathcal{Mod}(\sigma_2)(\mathcal{N}')$. Puisque $\mathcal{SI}$ est fermée par isomorphismes, $\mathcal{M}' \in \mathcal{Mod}(T_1)$ et $\mathcal{N}' \in \mathcal{Mod}(T_2)$. Finallement, puisque le diagramme ci-dessus est un diagramme d'amalgation faible, il existe un Σ'-modèle $\mathcal{P} \in \mathcal{Mod}(\Sigma')$ tel que $\mathcal{Mod}(\sigma_1')(\mathcal{P}) = \mathcal{M}'$ et $\mathcal{Mod}(\sigma_2')(\mathcal{P}) = \mathcal{N}'$. Par la condition de satisfaction, on a alors $\mathcal{P} \in \mathcal{Mod}(\mathcal{Sen}(\sigma_1')(T_1) \cup \mathcal{Sen}(\sigma_2')(T_2))$. $\square$

Troisième partie

Combinaison de logiques

Ainsi que nous l'évoquions en introduction, il existe aujourd'hui une large variété de systèmes logiques pour le développement de systèmes informatiques de qualité (*cf.* [141]). Chacun d'eux a été défini dans un but précis (modélisation du temps, de la croyance, des types de données complexes, *etc.*) et est dédié à un domaine d'application particulier (évolution temporelle d'une base de données, comportement d'un système au cours du temps, modélisation de l'intelligence, *etc.*). La combinaison de logiques est une méthode permettant de gérer cette diversité.
D'un point de vue théorique, l'avantage à tirer de la combinaison de logiques est double. D'une part cela permet de voir des logiques existantes comme combinaison de logiques connues, ces dernières pouvant être plus simples, plus largement étudiées ou mieux outillées. D'autre part, cela permet de construire des logiques nouvelles à des fins spécifiques à partir de logiques existantes. Dans les deux cas, et sous réserve de disposer des résultats de préservation adéquats, il est possible de transférer des résultats, propriétés et outils des logiques sources vers la logique combinée, modulo d'éventuelles modifications mineures[4]. Ceci est particulièrement intéressant pour les résultats sous-jacents à la structuration et au raffinement des spécifications, à la complétude et à la correction du système de preuve associé, sa décidabilité, *etc.*, ainsi que pour des outils tels que les vérificateurs de modèles, les outils d'aide à la preuve, les générateurs de jeux de tests ou de code, les prototypeurs rapides, *etc.*
D'un point de vue applicatif, même si la multitude existante de formalismes permet de choisir pour chaque problème le formalisme le mieux adapté (que ce soit en termes de paradigme de spécification, de goûts et habitudes des utilisateurs, d'outillage, *etc.*), elle constitue plutôt un frein dans la pratique car elle remet en cause la nécessaire modularité des systèmes. En particulier, la réutilisation des spécifications, tout comme la notion de « composants sur l'étagère », ne peut être assurée sans une étude rigoureuse des liens entre systèmes logiques et des conditions de transfert d'un formalisme à un autre des spécifications et des propriétés qui leurs sont attachées (conséquence sémantique, correction, complétude, *etc.*).

Dans la pratique, la diversité existante de systèmes logiques est absolument nécessaire pour répondre aux besoins spécifiques des différents utilisateurs de formalismes de spécifications : logiciens, philosophes, linguistes, ingénieurs logiciels, *etc.* Ces logiques ont parfois été définies par combinaison de logiques connues. C'est le cas par exemple de la temporalisation de logiques étudiée par M. Finger et D. Gabbay dans [53], ou encore la combinaison de deux logiques modales pour obtenir une logique bi-modale (*cf.* [54, 58]). Les liens avec la (les) logique(s) source(s) étant évident, il est alors « facile » d'étudier les possiblités de transferts de propriétés, résultats et outils de(s) la logique(s) source(s) vers la nouvelle logique. Malheureusement, la situation n'est pas toujours celle que nous venons de décrire. En effet, il arrive qu'une logique

4. Par exemple, la preuve de l'existence d'un objet initial dans une classe d'algèbres (partielles, totales, mono/multi-sortes, *etc.*) définie par un ensemble de formules conditionnelles repose toujours sur le même principe. Il faut définir des congruences de façon à ce que les noyaux des fonctions d'interprétation des termes en constituent une et montrer ensuite que l'algèbre des termes quotientée par l'intersection de ces noyaux est un modèle. Seuls les détails propres au type d'algèbre considéré changent. Il y a donc fort à gagner à donner une preuve logique-indépendante et de l'instantier.

soit créée « *ex-nihilo* », *i.e.* en ne s'appuyant pas sur la définition d'une (ou plusieurs) logique(s) connue(s). Plus exactement, on peut vouloir créer une logique en incluant un à un les différents concepts désirés (variables, symboles de fonctions, de prédicats, de sortes, quantificateurs, modalités, *etc.*) sans se soucier d'en respecter la définition usuelle. Ce n'est qu'une fois cette nouvelle logique totalement définie que l'on constate éventuellement qu'elle peut être vue comme une combinaison de logiques existantes modulo des transformations casuelles (*cf.* l'extension de la logique du premier ordre aux aspects dynamiques présentée en section 2.1).
Cet aspect de la combinaison de logiques est mal pris en compte par les travaux existants qui se cantonne à l'étude de simple combinaison de logiques existantes. De fait, les travaux existants (du moins ceux dont nous avons connaissance) sur la combinaison de logiques ne fournissent pas à proprement parler de solutions pour montrer qu'une logique quelconque est une combinaison de logiques existantes mais plutôt des conditions permettant le transfert et la préservation de propriétés entre deux logiques données et leur combinaison.

Le centre d'intérêt principal de cette thèse est l'étude des conditions suffisantes permettant de montrer l'interpolation de Craig dans le cadre des institutions. Nous nous intéressons alors dans cette troisième partie à la préservation de l'interpolation de Craig au travers de notre définition de la combinaison de logiques. Pour ce faire, nous nous plaçons dans le cadre des κ-institutions stratifiées et proposons en premier lieu une nouvelle définition de la combinaison de logiques étayée par la présentation d'un exemple d'introduction de logique (une restriction de la logique introduite dans [2]). Nous formulons ensuite des conditions permettant de préserver l'interpolation de Craig et la consistance de Robinson au travers de cette combinaison.

Chapitre 1

Combinaison de systèmes logiques

1.1 Combinaison dans les institutions

Il est bien connu qu'en théorie des catégories, le concept de flèche est plus fondamentale que celui d'objet [1]. La théorie des institutions fournit un bel exemple de ceci. En effet, il est possible, en prenant les institutions comme objets, de définir des catégories différentes suivant la notion de flèche adoptée. Nous définissons ici celle obtenue en considérant les morphismes d'institutions pour flèches. Cette catégorie posède la propriété remarquable d'être complète. Ainsi les limites, pour un foncteur dont le domaine est la catégorie des institutions représentent effectivement la notion de combinaison de différentes institutions.

Définition 1.1.1 *Soient* $\mathcal{I} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, \models)$, $\mathcal{I}' = (\mathit{Sig}', \mathit{Sen}', \mathit{Mod}', \models')$ *et* $\mathcal{I}'' = (\mathit{Sig}'', \mathit{Sen}'', \mathit{Mod}'', \models'')$ *trois institutions. Soient* $\mu_1 : \mathcal{I}'' \to \mathcal{I}'$ *et* $\mu_2 : \mathcal{I}' \to \mathcal{I}$ *deux morphismes d'institutions. Leur* composé $\mu = \mu_2 \circ \mu_1 : \mathcal{I}'' \to \mathcal{I}$ *est défini de la façon suivante :*

- $\Phi = \Phi_2 \circ \Phi_1 : \mathit{Sig}'' \to \mathit{Sig}$;
- $\alpha = \alpha_1 \circ (\Phi_1.\alpha_2) : \mathit{Sen} \circ \Phi_2 \circ \Phi_1 \to \mathit{Sen}''$;
- $\beta = ((\Phi_1)^{op}.\beta_2) \circ \beta_1 : \mathit{Mod}'' \to \mathit{Mod} \circ (\Phi_2 \circ \Phi_1)$.

Ajouter les figures !

Bien entendu, pour toute institution $\mathcal{I} = (\mathit{Sig}, \mathit{Sen}, \mathit{Mod}, \models)$, il existe un morphisme identité $\mu : \mathcal{I} \to \mathcal{I}$. Ce morphisme est simplement le morphisme dont toutes les composantes sont l'identité. Ceci définit alors une catégorie dont les objets sont les institutions et les flèches les morphismes d'institutions. On la note Ins.

Théorème 1.1.1 ([130]) *La catégorie* Ins *des institutions et des morphismes d'institutions est complète.*

1. Il est en effet toujours possible de définir une catégorie uniquement en termes de flèche, les objets étant alors identifier aux flèches identités (*cf.* [78]).

Exemple 1.1.1 ([130]) *Considérons les deux extensions suivantes de l'institution* EQL *de la logique équationnelle (cf. 1.1.6) :*

- EQL^{inh} *est l'extension de* EQL *obtenue en ajoutant le prédicat indiquant si l'ensemble de base d'une sorte donnée est ou non vide. On a alors le morphisme évident* $\mu_{inh} : \mathrm{EQL}^{inh} \to \mathrm{EQL}$ *;*
- EQL^{onto} *est l'extension de* EQL *obtenue en ajoutant le prédicat indiquant qu'un symbole de fonction donné est surjectif*[2]*. On a alors le morphisme évident* $mu_{onto} : \mathrm{EQL}^{onto} \to \mathrm{EQL}$.

Il est aisé de construire la pullback suivant de ces deux extensions de EQL *dans la catégorie Ins :*

$$\begin{tikzcd}
 & \mathrm{EQL}^{inh+onto} \arrow[dl] \arrow[dr] & \\
\mathrm{EQL}^{inh} \arrow[dr, "\mu_{inh}"'] & & \mathrm{EQL}^{onto} \arrow[dl, "\mu_{onto}"] \\
 & \mathrm{EQL} &
\end{tikzcd}$$

L'institution pullback $\mathrm{EQL}^{inh+onto}$ *possède les mêmes signatures et modèles que l'institution* EQL *(ainsi que* EQL^{inh} *et* EQL^{onto}*). Les formules de* $\mathrm{EQL}^{inh+onto}$ *sont, soit les équations usuelles (avec les translations le long des morphismes de signatures et la relation de satisfaction héritées de* EQL*), soit le prédicat de « vide » (avec les translations le long des morphismes de signatures et la relation de satisfaction héritées de* EQL^{inh}*), soit enfin le prédicat de surjectivité (avec les translations le long des morphismes de signatures et la relation de satisfaction héritées de* EQL^{onto}*). Les morphismes évidents de* $\mathrm{EQL}^{inh+onto}$ *vers, respectivement,* EQL^{inh} *et* EQL^{onto} *forment un pullback de* μ_{inh} *et* μ_{onto} *dans Ins.*

Le théorème précédent montre que la combinaison de systèmes logiques représentés sous forme d'institutions peut effectivement se faire via la notion de limites dans la catégorie *Ins*. Intuitivement, les limites dans la catégorie *Ins* sont construites en prenant, d'une part les limites des catégories de signatures et de modèles et, d'autre part les colimites des ensembles de formules. On remarque tout de suite qu'alors seuls les signatures et les modèles sont directement mélangés pour produire des signatures et des modèles amalgamés. En effet, ce sont les ensembles de formules et non les constructeurs de formules eux-mêmes qui sont combinés par colimites dans *Set*. Le résultat est alors une union des ensembles de formules, *i.e.* il n'y a pas de formules mixtes.

Notons que la notion de comorphisme permet également de définir une catégorie, noté *coIns*. Bien que cete catégorie possède des propriétés intuitivement (mais non formellement) duales de celles de *Ins*, il ne semble pas possible de combiner des institutions dans la catégorie *coIns* (*cf.* [132]). Certains posent alors la question de savoir s'il est néanmoins possible de construire de façon systématique des comorphismes lorsque l'on combine des institutions au sein

2. Ceci est exprimable dans FOL mais pas dans EQL

d'une institution « universelle » fixée. Bien que constituant un axe de recherche intéressant ceci ne rentre pas dans le cadre de notre étude, la logique résultante étant alors une restriction particulière de cette institution universelle et non l'extension d'une logique à un (ou plusieurs) aspect(s) nouveau(x).

1.2 Un cadre abstrait pour la combinaison : les parchemins

Ainsi que nous l'avons vu dans la section 1.1, le cadre des institutions n'est pas suffisamment précis pour y définir la notion de combinaison de logiques. Ceci est du au fait que les institutions ne considèrent pas la structure des formules. Cependant, les parchemins, introduits par J. Goguen et R. Burstall, en introduisant une notion de syntaxe abstraite, permettent de préciser la structure des formules et donc de contourner le problème exposé à la fin de la section précédente. Différentes notions de parchemins ont été introduites de façon à, d'une part palier à certains problèmes fondationnels de la défintion originelle des parchemins et, d'autre part mieux prendre en compte les spécificités de la fibration. Nous présentons ici la plus significative de ces notions de parchemins :les parchemins modèles-théoriques.

1.2.1 Définitions et exemples

Dans [97], T. Mossakowski, A. Tarlecki et W. Pawlowski ont proposé une notion légèrement différente de parchemins, appelé λ-parchemins et se sont intéressés au problème de la combinaison de systèmes logiques dans ce cadre. La raison pour avoir introduit les λ-parchemins est que les parchemins au sens de [62] repose sur les concepts de signature interne « universelle » et de structure sémantique « universelle » dont l'existence, comme on peut le voir sur des exemples très simples (*cf.* [62]), est questionnable. Dans les λ-parchemins, les modèles ont été séparés des signatures et la signature et la structure sémantique « universelles » ont été déplacées du niveau interne de la logique à celui de sa présentation. Dans [98], les auteurs continuent sur la même voie en introduisant les parchemins modèles-théoriques. Ces derniers modifient les λ-parchemins en distribuant la structure sémantique universelle sur toutes les signatures et tous les modèles de la logique.

Notation 1.2.1 *On introduit les notations suivantes :*

- $\mathcal{AlgSig}$ *est la catégorie des signatures du premier ordre multi-sortes* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ *;*
- $\mathcal{Logic} \in |\mathcal{AlgSig}|$ *est la signature du premier ordre multi-sorte réduite à une sorte* $*$ *et un symbole de prédicat* $D : *$*, i.e.* $\mathcal{Logic} = (\{*\}, \varnothing, \{D\})$ *;*
- $\mathcal{AlgSig}_*$ *est la sous-catégorie de* $\mathcal{AlgSig}$ *dont tous les objets contiennent* $\mathcal{Logic}$ *;*
- *pour toute signature* Σ*,* $\mathcal{Str}(\Sigma)$ *est la catégorie des* Σ*-structures du premier ordre.*

Définition 1.2.1 (parchemin (modèle théorique)) *Un* parchemin modèle-théorique *est la donnée d'un quadruplet* $\mathcal{P} = (\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *tel que :*

- $\mathcal{Sig}$ *est une catégorie de* signatures *;*

- $\mathcal{Mod} : \mathcal{Sig}^{op} \to \mathcal{Class}$ *est un* foncteur, *où* $\mathcal{Class}$ *est la catégorie des classes*[3] *;*
- $L : \mathcal{Sig} \to \mathcal{AlgSig}_*$ *est un* foncteur *associant une syntaxe abstraite* $L(\Sigma) \in |\mathcal{AlgSig}_*|$ *à toute signature* $\Sigma \in |\mathcal{Sig}|$ *;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *et tout* Σ*-modèle* $\mathcal{M} \in \mathcal{Mod}(\Sigma)$, $\mathcal{G}_\Sigma(\mathcal{M}) \in |\mathcal{Str}(L(\Sigma))|$ *est une* $L(\Sigma)$*-structure déterminant l'évaluation sémantique des* Σ*-formules dans* $\mathcal{M}$ *;*
- *pour tout morphisme de signatures* $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$ *et tout* Σ'*-modèle* $\mathcal{M}' \in \mathcal{Mod}(\Sigma')$, $\mathcal{G}_\sigma(\mathcal{M}') : \mathcal{G}_\Sigma(\mathcal{Mod}(\sigma)(\mathcal{M}')) \to L(\sigma)(\mathcal{G}_{\Sigma'}(\mathcal{M}'))$ *est un* $L(\Sigma)$*-morphisme tel que :*
  $\forall \sigma' \in \mathcal{Sig}(\Sigma'', \Sigma'), \forall \sigma'' \in \mathcal{Sig}(\Sigma, \Sigma'')$,
  $$\mathcal{G}_{\sigma' \circ \sigma''}(\mathcal{M}') = L(\sigma'')(\mathcal{G}_{\sigma'}(\mathcal{M}')) \circ \mathcal{G}_{\sigma''}(\mathcal{Mod}(\sigma')(\mathcal{M}'))$$

Pour toute signature Σ et tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, l'ensemble $|\mathcal{G}_\Sigma(\mathcal{M})|_*$ est l'espace des valeurs logiques du parchemin dans $\mathcal{M}$. Dans le cas d'une logique bi-valuée, pour tout modèle $\mathcal{M}$, on a alors $|\mathcal{G}_\Sigma(\mathcal{M})|_* = \{0, 1\}$ mais il est également possible de capturer des logiques multi-valuées dans le cadre des parchemins modèles-théoriques. Dans ce dernier cas, on retrouve la notion ultime de deux valeurs de vérité par le prédicat unaire $D_{\mathcal{G}_\Sigma(\mathcal{M})} \subseteq |\mathcal{G}_\Sigma(\mathcal{M})|_*$ qui sépare les valeurs logiques désignées, et donc interprétées comme vraies, des autres valeurs, interprétées come fausses. La condition imposée par le dernier point de la définition ci-dessus induit que la désignation des valeurs logiques est préservée le long des morphisme de signatures. Elle est reflétée[4] si le parchemin considéré vérifie la condition supplémentaire indiqué dans la définition suivante.

Définition 1.2.2 (parchemin logique) *Un parchemin* $\mathcal{P} = (\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *est dit* logique *si pour tout morphisme de signatures* $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$ *et pour tout* Σ'*-modèle* $\mathcal{M}' \in \mathcal{Mod}(\Sigma')$, *le* $L(\Sigma)$*-homomorphisme* $\mathcal{G}_\sigma(\mathcal{M}') : (\mathcal{Mod}(\sigma)(\mathcal{M}')) \to L(\sigma)(\mathcal{G}_{\Sigma'}(\mathcal{M}'))$ *est un monomorphisme extreme*[5].

Un parchemin modèle-théorique $\mathcal{P} = (\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ présente l'institution $\mathcal{I}(\mathcal{P}) = (\mathcal{Sig}, \mathcal{Sen}, \mathcal{Mod}, \models)$ telle que :
- pour toute signature $\Sigma \in |\mathcal{Sig}|$, $\mathcal{Sen}(\Sigma) = |T_{L(\Sigma)}|_*$, où $|T_{L(\Sigma)}|$ est la $L(\Sigma)$-algèbre initiale ;
- pour tout morphisme de signatures $\sigma \in \mathcal{Sig}(\Sigma, \Sigma')$, $\mathcal{Sen}(\sigma : \Sigma \to \Sigma') = (T_{L(\sigma)})_*$, où $T_{L(\sigma)} : T_{L(\Sigma)} \to L(\sigma)(T_{L(\Sigma')})$ est l'homomorphisme initial ;
- pour toute signature $\Sigma \in |\mathcal{Sig}|$, tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$ et toute Σ-formule $\varphi \in \mathcal{Sen}(\Sigma)$, $\mathcal{M} \models_\Sigma \varphi$ si et seulement si $\overline{\mathcal{M}}_*(\varphi) \in D_{\mathcal{G}_\Sigma(\mathcal{M})}$, où $\overline{\mathcal{M}} : T_{L(\Sigma)} \to \mathcal{G}_\Sigma(\mathcal{M})$ est l'homomorphisme initial.

Exemple 1.2.1 (logique propositionnnelle) *Le parchemin de la logique propositionnelle est la donnée du quadruplet* $(\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *tel que :*
- $\mathcal{Sig}$ *et* $\mathcal{Mod}$ *sont définis comme dans l'exemple 1.1.1, partie I ;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$, $L(\Sigma)$ *est la signature du premier ordre multi-sorte* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ *telle que :*
  - $\mathcal{S} = \{*\}$,

3. *cf.* annexe A pour la définition de la catégorie $\mathcal{Class}$. Intuitivement, ceci signifie qu'on ne considère pas ici les morphismes de modèles.
4. La reflection exprime la notion duale de la préservation
5. *cf.* annexe A

  - $\mathcal{F}_* = \{p : */p \in |\Sigma|\}$,
  - $\mathcal{F}_{**} = \{\neg\}$,
  - $\mathcal{F}_{*^2*} = \{\vee\}$,
  - $\mathcal{R}_* = \{D : *\}$.
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *et tout* Σ*-modèle* $v \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{G}_\Sigma(v) \in |\mathcal{Str}(L(\Sigma))|$ *est définie de la façon suivante :*
  - $|\mathcal{G}_\Sigma(v)|_* = \{0, 1\}$, $D_{\mathcal{G}_\Sigma(v)} = \{1\}$,
  - $\forall p \in \Sigma, p^{\mathcal{G}_\Sigma(v)} = v(p)$,
  - $\forall b \in |\mathcal{G}_\Sigma(v)|_*$, $\neg b = 1 - b$,
  - $\forall b_1, b_2 \in |\mathcal{G}_\Sigma(v)|_*$, $b_1 \vee b_2 = \mathtt{sup}(b_1, b_2)$.

Exemple 1.2.2 (EQL$^-$) *Le parchemin de la logique équationnel sans égalité est la donnée du quadruplet* $(\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *tel que :*

- $\mathcal{Sig}$ *et* $\mathcal{Mod}$ *sont définis comme dans l'exemple 1.1.6, partie I ;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$, $L(\Sigma) = \Sigma \cup \mathcal{Logic}$ *;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *et tout* Σ*-modèle* $v \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{G}_\Sigma(v) \in |\mathcal{Str}(L(\Sigma))|$ *est définie de la façon suivante :*
  - $|\mathcal{G}_\Sigma(\mathcal{M})|_* = \{0, 1\}$, $D_{\mathcal{G}_\Sigma(v)} = \{1\}$,
  - *pour tout* $s \in S$, $|\mathcal{G}_\Sigma(\mathcal{M})|_s = M_s$,
  - *pour tout symbole de fonction* $f \in F$, $f^{\mathcal{G}_\Sigma(\mathcal{M})} = f^{\mathcal{M}}$.

Exemple 1.2.3 (EQL) *Le parchemin de la logique équationnel est la donnée du quadruplet* $(\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *tel que :*

- $\mathcal{Sig}$ *et* $\mathcal{Mod}$ *sont définis comme dans l'exemple précédent ;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$, $L(\Sigma) = L^{\mathrm{EQL}^-}(\Sigma)$ *plus* $=: s \times s \to *$ *pour tout* $s \in S$ *(où* S *est l'ensemble de sortes de la signature* Σ*) ;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *et tout* Σ*-modèle* $v \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{G}_\Sigma(v) \in |\mathcal{Str}(L(\Sigma))|$ *est définie de la façon suivante :*
  - $\mathcal{G}_\Sigma(\mathcal{M})_{|L^{\mathrm{EQL}^-}} = \mathcal{G}_\Sigma^{\mathrm{EQL}^-}(\mathcal{M})$,
  - $t =_{\mathcal{G}_\Sigma(\mathcal{M})} t' = 1$ *si* $t = t'$ *et* 0 *sinon.*

Exemple 1.2.4 (PA$^-$) *Le parchemin des algèbres partielles sans égalité est la donnée du quadruplet* $(\mathcal{Sig}, \mathcal{Mod}, L, \mathcal{G})$ *tel que :*

- $\mathcal{Sig}$ *et* $\mathcal{Mod}$ *sont définis comme dans l'exemple 1.1.10, partie I ;*
- *pour toute signature* $\Sigma = (\mathcal{S}, \mathcal{PF}, \mathcal{TF}) \in |\mathcal{Sig}|$, $L(\Sigma) = L^{\mathrm{EQL}^-}(\mathcal{S}, \mathcal{PF} \cup \mathcal{TF})$ *;*
- *pour toute signature* $\Sigma \in |\mathcal{Sig}|$ *et tout* Σ*-modèle* $v \in |\mathcal{Mod}(\Sigma)|$, $\mathcal{G}_\Sigma(v) \in |\mathcal{Str}(L(\Sigma))|$ *est définie de la façon suivante :*
  - $|\mathcal{G}_\Sigma(\mathcal{M})|_* = \{0, 1\}$, $D_{\mathcal{G}_\Sigma(\mathcal{M})} = \{1\}$,
  - *pour tout* $s \in S$, $|\mathcal{G}_\Sigma(\mathcal{M})|_s = M_s \cup \{\bot\}$, *où* $\bot$ *est un élément nouveau particulier,*
  - *pour tous symboles de fonctions* $Tf \in \mathcal{TF}$ *et* $Pf \in \mathcal{PF}$, $Tf^{\mathcal{G}_\Sigma(\mathcal{M})}$ *et* $Pf^{\mathcal{G}_\Sigma(\mathcal{M})}$ *sont les extensions respectives usuelles de* $Tf^{\mathcal{M}}$ *et* $Pf^{\mathcal{M}}$ *sur les ensembles de base de* $\mathcal{M}$ *auxquels on a ajouté l'élément indéfini* $\bot$.

Il est possible de remonter le concept de morphisme d'institutions au niveau des parchemins modèle-théoriques de façon à capturer le fait qu'un parchemin est construit au-dessus d'un autre.

Définition 1.2.3 (morphisme) *Un morphisme de parchemins modèles-théoriques* $(\Phi, \mu) : \mathcal{P} = (\mathcal{S}ig, \mathcal{M}od, L, \mathcal{G}) \to \mathcal{P}' = (\mathcal{S}ig', \mathcal{M}od', L', \mathcal{G}')$ *est la donnée :*

- *d'un foncteur* $\Phi : \mathcal{S}ig \to \mathcal{S}ig'$ *;*
- *d'une transformation naturelle* $\alpha : L' \circ \Phi \Rightarrow L$ *;*
- *d'une transformation naturelle* $\beta : \mathcal{M}od \Rightarrow \mathcal{M}od' \circ \Phi^{op}$ *;*
- *d'une 2-transformation naturelle* $g : (\mathcal{G}' * \Phi^{op}) \circ \beta \Rightarrow (\mathcal{S}tr * \alpha^{op}) \circ \mathcal{G}$*, i.e. pour toute signature* $\Sigma \in |\mathcal{S}ig|$*, une transformation naturelle* $g_\Sigma : \mathcal{G}'_{\Phi(\Sigma)} \circ \beta_\Sigma \Rightarrow \mathcal{S}tr(\alpha_\Sigma) \circ \mathcal{G}_\Sigma$*. On obtient donc, pour tout* Σ*-modèle* $\mathcal{M} \in \mathcal{M}od(\Sigma)$*, un* $L'(\Phi(\Sigma))$*-homomorphisme* $g_{\Sigma,\mathcal{M}} : \mathcal{G}'_{\Phi(\Sigma)}(\beta_\Sigma(\mathcal{M})) \to \mathcal{G}_{\Sigma|\alpha_\Sigma}(\mathcal{M})$ *tel que pour tout morphisme de signature* $\sigma \in \mathcal{S}ig(\Sigma_1, \Sigma_2)$ *et tout* Σ_2*-modèle* $\mathcal{M}_2 \in \mathcal{M}od(\Sigma_2)$*,* $\mathcal{G}_{\sigma|\alpha_{\Sigma_1}}(\mathcal{M}_2) \circ g_{\Sigma_1,\mathcal{M}_2|\sigma} = g_{\Sigma_2,\mathcal{M}_2|L'(\Phi(\sigma))} \circ \mathcal{G}'_{\Phi(\sigma)}(\beta_{\Sigma_2}(\mathcal{M}_2))$.

Chaque $g_{\Sigma,\mathcal{M}}$ préserve les valeurs de vérité désignées mais ne les reflète pas nécessairement.
Un morphisme de parchemins modèles-théoriques $\mu = (\Phi, \alpha, \beta, g) : \mathcal{P} \to \mathcal{P}'$ est dit *logique* si pour toute signature $\Sigma \in |\mathcal{S}ig|$ et tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$ dans $\mathcal{P}$, $g_{\Sigma,\mathcal{M}}$ est un monomorphisme extreme (*i.e.* une injection close [**À préciser !**]). Les parchemins logiques constituent alors les objets d'une catégorie dont les flèches sont les morphismes logiques (*cf.* [98]).

Une condition naturelle pour combiner des parchemins modèle-théoriques est d'imposer la préservation des valeurs de vérité de façon à ce que la combinaison de parchemins représente effectivement la combinaison des institutions correspondantes. Ceci signifie que l'on souhaite prendre des limites dans la catégorie des parchemins logiques munie des morphismes logiques. Malheureusement, ceci n'est pas toujours possible car cette dernière catégorie n'est pas complète (*cf.* [98]). Par contre, la catégorie des parchemins munie des morphismes de parchemins est complète ([98]).

Exemple 1.2.5 ([98]) *Soit la combinaison suivante de* EQL *et* PA^- *via un pullback dans la catégorie des parchemins :*

$$\begin{tikzcd}
\mathrm{PA} \arrow[r] \arrow[d] & \mathrm{PA}^- \arrow[d] \\
\mathrm{EQL} \arrow[r] & \mathrm{EQL}^-
\end{tikzcd}$$

Alors, le parchemin PA *est tel que :*

- *les signatures et les modèles sont ceux de* PA^- *comme attendu ;*
- *le langage sur une signature* $(\mathcal{S}, \mathcal{TF}, \mathcal{PF})$ *coincide avec le langage de* EQL *sur* $(\mathcal{S}, \mathcal{TF} \cup \mathcal{PF})$*. Les formules générées continnent donc les égalités entre* $(\mathcal{S}, \mathcal{TF}, \mathcal{PF})$*-termes comme attendu ;*
- *les structures sémantiques coincident avec celle de* PA^- *sauf en ce qui concerne leurs réduits à la sous-signature* $\mathcal{L}ogic$*. Pour toute signature* $\Sigma = (\mathcal{S}, \mathcal{TF}, \mathcal{PF})$ *et tout* Σ*-modèle* $\mathcal{M} \in |\ \mathrm{mod}\ (\Sigma)|$*, l'interprétation des égalités* $=_{G_\Sigma(\mathcal{M})}$ *sur les éléments de* M_s *est hérité de* EQL *comme*

attendu. Cependant, $|G_\Sigma(\mathcal{M})|_*$ *dans* PA *étend l'ensemble* $\{0,1\}$ *par des nouvelles valeurs générées librement par les interprétations de* $t =_{G_\Sigma(\mathcal{M})} \bot$ *et* $\bot =_{G_\Sigma(\mathcal{M})} t$.

Le résultat de la combinaison de parchemins présentée ci-dessus n'est pas tout à fait celui attendu du fait de l'émergence de nouvelles valeurs logiques. Deux méthodes permettent de solutionner ce problème. La première solutions consiste à quotienter les structures sémantiques par une congruence qui identifierai les nouvelles valeurs avec l'une des deux valeurs de vérité définies (*i.e.* avec 0 ou 1). Cette possibilité a d'ailleurs déjà fait l'objet de discussions dans le cadre de la définition originelle des parchemins (*cf.* [94]). UNe autre solution consiste à remonter au niveau des parchemins la notion de co-morphisme d'institutions et de combiner les parchemins dans un parchemin « universel » à l'aide de ces représentations (*cf.* [98] pour plus de détail). Cette dernière solution, quoique porteuse de certains résultats intéressants ne coincide pas avec notre vision de la combinaison de logique, *i.e.* la création de logiques via la combinaison de logiques connues moyennant certaines transformations des logiques sources.

1.2.2 Extraction à partir d'une κ-institution stratifiée

Une κ-institution stratifiée $\kappa\mathcal{SI} = (\mathcal{S}ig, \kappa, \mathcal{M}od, [\![]\!])$ présente le parchemin modèle-théorique $\mathcal{P}(\kappa\mathcal{SI}) = (\mathcal{S}ig, \mathcal{M}od, L, \mathcal{G})$ tel que :

- pour toute signature $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R}) \in |\mathcal{S}ig|$, $L(\Sigma) = (S, F, R)$ est définie de la façon suivante :
  - $S = \mathcal{S} \cup \{*\}$,
  - $F_{*^n} = \kappa(\Sigma)^n$,
  - pour tout $s \in \mathcal{S}$, $F_s = T_\Sigma(\mathcal{X})_s$,
  - pour tout $\omega \in \mathcal{S}^*$, $F_{\omega *} = \mathcal{R}_\omega$
  - $R_* = \{D : *\}$.
- pour toute signature $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R}) \in |\mathcal{S}ig|$ et tout Σ-modèle $\mathcal{M} \in |\mathcal{M}od(\Sigma)|$, $\mathcal{G}_\Sigma(\mathcal{M})$ est définie de la façon suivante :
  - $|\mathcal{G}_\Sigma(v)|_* = \{0,1\}$, $D_{\mathcal{G}_\Sigma(v)} = \{1\}$,
  - pour tout $s \in \mathcal{S}$, $|\mathcal{G}_\Sigma(\mathcal{M})|_s = M_s$,
  - pour tout symbole de fonction $f \in \mathcal{F}$, $f^{\mathcal{G}_\Sigma(\mathcal{M})} = f^{\mathcal{M}}$,
  - pour tout symbole de prédicat $r \in \mathcal{R}$, $r^{\mathcal{G}_\Sigma(\mathcal{M})} = r^{\mathcal{M}}$,
  - pour tout constructeur de formule $c \in F_{*^n *}$, pour tout n-uplet $(b_1, \ldots, b_n) \in |\mathcal{G}_\Sigma(\mathcal{M})_*|$, $c(b_1, \ldots, b_n) = c^{[\![\mathcal{M}]\!]}(b_1, \ldots, b_n)$.

Chapitre 2

Combinaison de κ-institutions stratifiées

Nous introduisons dans ce chapitre une nouvelle définition de la combinaison de logiques. La raison principale pour introduire cette nouvelle définition est l'inadéquation des définitions existantes de la combinaison de logiques pour prendre en compte l'introduction de formalismes. Au niveau syntaxique, en effet, la combinaison de logiques consiste simplement en une union des constructeurs de formules. Cependant, et ainsi que nous le verrons dans l'exemple introductif, un formalisme de spécification se définit souvent comme la combinaison de formalismes existants modulo certaines transformations. Ce sont ces dernières qui ne sont pas prisent en compte par les définitions existantes de la combinaison de logiques. Nous les représenterons ici par la notion de morphismes d'institutions.

2.1 Un exemple introductif

2.1.1 Définition du formalisme

Dans [2], M. Aiguier a introduit le formalisme des ÉTOILE-spécifications pour la spécification de systèmes dynamiques et temps-réel. Ce formalisme est une extension des spécifications algébriques classiques où les notions d'état et de temps sont traités de façon implicite à l'instar des logiques modales. Le comportement dynamique des systèmes n'est alors observé qu'au travers des opérations (également appelées méthodes) disponibles dans le système. Nous ne présentons pas ici le formalisme des ÉTOILE-spécifications dans son ensemble mais plutôt une restriction aux seuls aspects dynamiques. Le lecteur intéressé pourra se reporter à [3, 4] pour une présentation complète de ce formalisme.

Syntaxe

Nous introduisons dans ce paragraphe les notions de signatures et de formules construites au-dessus d'une signature donnée.

Définition 2.1.1 (signature) *Une* signature *est la donnée d'un couple* $(\mathcal{S}, \mathcal{F})$ *où* $\mathcal{S}$ *est un ensemble de sortes et* $\mathcal{F}$ *est un ensemble de noms de fonctions, chacun muni d'un profil dans* $\mathcal{S}^* \times \mathcal{S} \cup \{\epsilon\}$. *On notera* $f : s_1 \times \cdots \times s_n \to s$ *le* *fait que $f \in \mathcal{F}$ est muni du profil $(s_1 \ldots s_n, s) \in \mathcal{S}^* \times \mathcal{S}$ et $f : s_1 \times \cdots \times s_n \to \epsilon$ pour signifier que $f \in \mathcal{F}$ est muni du profil $(s_1 \ldots s_n, \epsilon) \in \mathcal{S}^* \times \{\epsilon\}$.*

Une opération $f : s_1 \times \cdots \times s_n \to \epsilon$ représente une méthode dont l'action sera de modifier le système sans observer une valeur dessus. Par analogie avec les langages de programmation impératifs, f est le nom d'une procédure, *i.e.* une application rendant un état comme résultat et aucune valeur. À l'inverse, une fonction de profil $s_1 \times \cdots \times s_n \to s$ est le nom d'une fonction, *i.e.* une application rendant à la fois comme résultat une valeur du type de données s et un état (on prends ainsi en compte la notion d'effet de bord).

Exemple 2.1.1 *Une signature pour la spécifications de piles dynamiques est le couple $(\mathcal{S}, \mathcal{F})$ définit de la façon suivante :*

- $\mathcal{S} = \{\mathtt{nat}, \mathtt{elem}\}$;
- $\mathcal{F} = \{\mathtt{push} : \mathtt{elem} \to \epsilon; \mathtt{pop\_top} :\to \mathtt{elem}\}$.

Intuitivement dans l'exemple ci-dessus, c'est le contenu d'une pile qui détermine son état.

Un *morphisme* de signatures $\mu : \Sigma \to \Sigma'$ est alors la donnée de deux applications (μ_1, μ_2) où μ_1 est définie sur les sortes et μ_2 sur les noms d'opérations et telles que le profil de chacune des opérations est préservé par μ_1. Ceci définit clairement une catégorie que l'on notera $\mathcal{S}ig_*$.

Avant de définir les formules il faut, comme pour tout formalisme non-propositionnel, définir l'ensemble des termes avec variables. Étant donné une signature $\Sigma \in |\mathcal{S}ig_*|$, on considère alors un ensemble de variables $\mathcal{X}$ défini comme un ensemble muni d'une partition indexée par l'ensemble de sortes $\mathcal{S}$ (on parle également de $\mathcal{S}$-ensemble). Ce qui différencie ici les termes de ceux usuellement définis dans les formalismes algébriques classiques est justement la prise en compte de la dynamique des systèmes. En effet, nous devons être capables d'établir des identités entre états et entre valeurs. Comme chaque opération du système peut modifier l'état du système, l'ordre dans lequel elles sont exécutées est important. On retrouve alors les deux ordres naturels d'exécution : séquentiel et indifférencié. Enfin, on doit être capable d'établir le fait qu'une fonction ne possède pas d'effet de bord.

Définition 2.1.2 (terme) *Soit $\Sigma = (\mathcal{S}, \mathcal{F})$ une signature. On définit les différents types de termes suivants :*

- *_ de sorte ϵ. Ce terme caractérise l'action de ne rien faire sur le système ;*
- *$t_1; t_2$ et t_1, t_2 de sorte $\alpha.\beta \in \mathcal{S}^*$, où α est la sorte de t_1 et β celle de t_2. Ceci représente respectivement l'ordre séquentiel et l'ordre indifférencié des deux actions représentées par les termes t_1 et t_2 ;*
- *$f(t)$ de sorte s où $f : s_1 \times \cdots \times s_n \to s$ et t est de sorte $s_1 \ldots s_n$. Ce terme caractérise l'application de l'opération f sur le n-uplet résultant de l'évaluation de t.*

On note $T_\Sigma(\mathcal{X})$ la famille indexée par $\mathcal{S}^*$ des termes avec variables dans $\mathcal{X}$ définis sur la signature Σ suivant les règles données ci-dessus.

On peut maintenant définir les formules construites sur une signature donnée.

Définition 2.1.3 (formule) *Soit* $\Sigma = (\mathcal{S}, \mathcal{F})$ *une signature. On note* $\mathcal{Sen}_*(\Sigma)$ *le plus petit ensemble au sens de l'inclusion vérifiant :*

- $\{t = t'/t, t' \in T_{\Sigma,s}(\mathcal{X})\} \subseteq \mathcal{Sen}_*(\Sigma)$. *Ces formules définissent une identité entre deux valeurs de même sorte ;*
- $\{t \equiv t'/t, t' \in T_{\Sigma}(\mathcal{X})\} \subseteq \mathcal{Sen}_*(\Sigma)$. *Ces formules définissent une identité entre deux états ;*
- $\forall t \in T_{\Sigma}(\mathcal{X}), \forall \varphi \in \mathcal{Sen}_*(\Sigma)$, $\mathtt{after}[t]\varphi \in \mathcal{Sen}_*(\Sigma)$. *Ces formules exprime la fait que le formule* φ *est vraie pour tout état du système résultant de l'exécution de* t *;*

La définition du foncteur $\mathcal{Sen}_* : \mathcal{Sig} \to \mathcal{Sen}_*$ devient alors naturelle :

- à toute signature $\Sigma \in |\mathcal{Sig}|$, $\mathcal{Sen}_*$ associe l'ensemble $\mathcal{Sen}_*(\Sigma)$;
- à tout morphisme de signature $\sigma \in \mathcal{Sig}$, $\mathcal{Sen}_*$ associe son extension canonique aux termes et aux formules.

Exemple 2.1.2 *Nous reprenons la signature introduite à l'exemple 2.1.1. Une axiomatisation des piles dynamiques peut, par exemple, être la donnée des deux formules suivantes :*

1. $\mathtt{after}[\mathtt{push}(e)](\mathtt{pop\_top} = e)$
2. $\mathtt{push}(e); \mathtt{pop\_top} \equiv \_$

Dans l'exemple ci-dessus, tandis que le premier des deux axiomes spécifie que l'élément se trouvant au sommet de la pile est le dernier que l'on vient d'y ranger, le deuxième spécifie que les opérations d'empilement et de dépilement sont inverses l'une de l'autre.

Sémantique

Nous introduisons dans ce paragraphe les notions de modèle associé à une signature donnée et de satisfaction des formules construites au-dessus d'une signature par les modèles associés à cette dernière.

Définition 2.1.4 (modèle) *Soit* $\Sigma = (\mathcal{S}, \mathcal{F})$ *une signature. Un* Σ-modèle $\mathcal{M}$ *est la donnée :*

- *d'une famille d'ensembles* $(M_s)_{s \in \mathcal{S}}$ *indexée par les sortes avec pour convention* $M_\epsilon = \{\mathbb{1}\}$ *;*
- *d'un ensemble* $\underline{M}$ *non-vide représentant les états du système ;*
- *pour chaque symbole de fonction* $f : s_1 \times \cdots \times s_n \to s \in \mathcal{F}$, *de deux familles d'applications* $f^{\mathcal{M}}$ *et* $\underline{f}^{\mathcal{M}}$ *indexées par* $\underline{M}$ *telles que :*
  $\forall \eta \in \underline{M}$,
  $$f^{\mathcal{M}}_{\eta} : M_{s_1} \times \cdots \times M_{s_n} \to M_s$$
  et
  $$\underline{f}^{\mathcal{M}}_{\eta} : M_{s_1} \times \cdots \times M_{s_n} \to \underline{M}$$

Intuitivement, l'ensemble $\underline{M}$ ci-dessus est l'esnemble de tous les états possibles du système.

Exemple 2.1.3 *Nous reprenons la signature introduite à l'exemple 2.1.1. Soit L un ensemble quelconque représentant les éléments à empiler et L^* l'ensemble des mots finis sur L. On définit alors la famille M_s pour $s \in \mathcal{S}$ par : $M_{\mathtt{nat}} = \mathbb{N}$, $M_{\mathtt{elem}} = L \cup \{e\}$*[1]*. L'ensemble des états $\underline{M} = L^*$. Pour tout mot $\alpha \in L^*$, la sémantique des opérations et des éléments est la suivante :*

- $\mathtt{push}_\alpha^{\mathcal{M}} : L \to \{1\!\!1\}$

- $\underline{\mathtt{push}_\alpha^{\mathcal{M}}} : \begin{array}{l} L \to L^* \\ l \mapsto l.\alpha \end{array}$

- $\mathtt{pop\_top}_\alpha^{\mathcal{M}} = \left\{ \begin{array}{l} e \ si \ \alpha = \epsilon \\ l \ si \ \alpha = l.\beta \end{array} \right.$

- $\underline{\mathtt{pop\_top}_\alpha^{\mathcal{M}}} = \left\{ \begin{array}{l} \epsilon \ si \ \alpha = \epsilon \\ \beta \ si \ \alpha = l.\beta \end{array} \right.$

Définition 2.1.5 (morphisme de Σ-modèles) *Soit $\Sigma \in |\mathcal{Sig}_*|$ une signature. Un* morphisme de Σ-modèles *$\mu : \mathcal{M}_1 \to \mathcal{M}_2$ est la donnée d'une application $\mu : M_1 \to M_2$ compatible avec les sortes, i.e. $\mu(M_{1,s}) \subseteq M_{2,s}$ pour tout $s \in \mathcal{S}$, et d'une application $\underline{\mu} : \underline{M_1} \to \underline{M_2}$ telles que le tout est compatible avec l''interprétation des opérations, i.e. :*

- $\mu(f_\eta^{\mathcal{M}_1}(a)) = f_\eta^{\mathcal{M}_2}(\mu(a))$;
- $\underline{\mu}(\underline{f}_\eta^{\mathcal{M}_1}(a)) = \underline{f}_{\underline{\mu}(\eta)}^{\mathcal{M}_2}(\mu(a))$.

Étant donnée une signature $\Sigma \in |\mathcal{Sig}_*|$, on note $\mathcal{Mod}_*(\Sigma)$ la catégorie dont les objets sont les Σ-modèles et les flèches les morphismes de Σ-modèles munis de la composition habituelles.

On étend maintenant cette interprétation aux morphismes de signatures par la notion de foncteur d'oubli.

Définition 2.1.6 (foncteur d'oubli) *Soit $\sigma \in \mathcal{Sig}_*(\Sigma, \Sigma')$ un morphisme de signatures. Le* foncteur d'oubli *$\mathcal{Mod}_*(\sigma) : \mathcal{Mod}_*(\Sigma') \to \mathcal{Mod}_*(\Sigma)$ est défini de la façon suivante :*

- *pour tout Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}_*(\Sigma')|$, $\mathcal{Mod}_*(\sigma)(\mathcal{M}')$ est le Σ-modèle $\mathcal{M}$ tel que :*
  - *$\forall s \in \mathcal{S}$, $M_s = A_{\sigma(s)}$ et $\underline{M} = \underline{A}$,*
  - *$\forall f \in \mathcal{F}, \forall \eta \in \underline{M}$, $f_\eta^{\mathcal{M}} = \sigma(f)_\eta^{\mathcal{M}'}$ et $\underline{f}_\eta^{\mathcal{M}} = \underline{\sigma(f)}_\eta^{\mathcal{M}'}$.*
- *pour tout morphisme de Σ'-modèles $\mu : \mathcal{M}'_1 \to \mathcal{M}'_2$, $\mathcal{Mod}_*(\mu) : \mathcal{Mod}_*(\mathcal{M}'_1) \to \mathcal{Mod}_*(\mu) : \mathcal{Mod}_*(\mathcal{M}'_2)$ est le morphisme de Σ-modèles défini par toutes les restrictions de μ de la forme $\mu_s : M'_{1,s} \to M'_{2,s}$ pour tout $s \in \mathcal{S}$ et $\underline{\mu}$.*

1. Le symbole e permet de traiter le cas exceptionnel de l'application de l'opération `pop_top` sur la liste vide. La valeur rendue est alors l'élément e.

L'interprétation des termes est ici plus complexe que dans le cas usuel des spécifications algébriques du fait que les états sont des contextes intentionnels pour l'évaluation des termes, *i.e.* leur évaluation n'est pas fonctionnelle. En effet, chaque opération de la signature est interprétée à partir d'un état et chacune peut modifier l'état du système. Ainsi, à partir d'un état donné, l'état résultant de l'évaluation d'un terme sera différent selon l'ordre d'évaluation choisi. L'évaluation d'un terme est alors non-déterministe du fait des effets de bords des opérations. Le résultat de l'évaluation d'un terme n'est donc plus une simple valeur mais un couples (r, η) où r est un n-uplet de valeurs et η est l'état atteint par l'évaluation du terme.
En résumé, pour interpréter un Σ-terme t dans un Σ-modèle $\mathcal{M}$, il faut d'abord interpréter les variables de t dans M selon leur type. Pour ce faire, on se donne un état pour amorcer le première interprétation et on évalue selon tous les ordres possibles les termes de la forme $f(a)$, où a est un n-uplet de valeurs appelé *terme plat*, à partir de l'état résultant de l'évaluation du terme plat précédent jusqu'à l'obtention d'une valeur finale.

Définition 2.1.7 (évaluation d'un terme plat) *Soient $\Sigma \in |\mathcal{Sig}_*|$ une signature et $\mathcal{M} \in |\mathcal{Mod}_*(\Sigma)|$. Étant donnés un terme plat $f(a)$ et un état $\eta \in \underline{M}$, son évaluation à partir de η est le couple $(f_\eta^{\mathcal{M}}(a), \underline{f}_\eta^{\mathcal{M}}(a))$.*

Un pas d'évaluation d'un terme t plus général consiste donc à évaluer un sous-terme plat de t. Le choix du sous-terme plat à évaluer doit respecter l'ordre séquentiel représenté par « ; » lorqu'il apparait dans t. Le sous-terme plat choisi de t compatible avec l'ordre séquentiel s'appelle alors une *occurence possible.* L'évaluation de t consiste alors à évaluer pas-à-pas chaque occurence possible jusqu'à obtenir une valeur. Cette évaluation se définit comme la fermeture reflexive et transitive de la relation binaire $\mathcal{E} \subseteq T_\Sigma(M) \times \underline{M}$ introduite dans la définition suivante.

Définition 2.1.8 (étape d'évaluation) *On note $\mathcal{E}$ la relation binaire définie sur $\mathcal{E} \subseteq T_\Sigma(M) \times \underline{M}$ de la façon suivante: $(t, \eta)\mathcal{E}(t', \eta')$ si et seulement s'il existe une occurence possible ω dont l'évaluation à partir de η est (v, η') et t' est obtenu à partir de t en remplaçant ω par la valeur v.*

Définition 2.1.9 (valeur et état déterministe) *Un terme $t \in T_\Sigma(M)$ a une valeur déterministe v à partir de η signifie que dans $\mathcal{M}$, quelque soit (v', η_1) tel que $(t, \eta)\mathcal{E}^*(v', \eta_1)$, on a $v = v'$.*
Un terme $t \in T_\Sigma(M)$ possède un état déterministe *η' à partir de η signifie que dans $\mathcal{M}$, quelque soit (v, η_1) tel qu $(t, \eta)\mathcal{E}^*(v, \eta_1)$, on a $\eta' = \eta_1$.*

Définition 2.1.10 (satisfaction) *Soient $\Sigma \in |\mathcal{Sig}_*|$ une signature, $\mathcal{M} \in |\mathcal{Mod}_*(\Sigma)|$ un Σ-modèle, $\varphi \in \mathcal{Sen}_*(\Sigma)$ une Σ-formule, $\eta \in \underline{M}$ et $\nu : \mathcal{X} \to M$. La* satisfaction de φ par $\mathcal{M}$ pour ν et η, *notée $\mathcal{M} \models_{\nu,\eta} \varphi$, est définie de la façon suivante :*

- *si φ est de la forme $t = t'$, alors $\mathcal{M} \models_{\nu,\eta} \varphi$ si $\nu^\sharp(t)$ et $\nu^\sharp(t')$ ont la même valeur déterminste ;*
- *si φ est de la forme $t \equiv t'$, alors $\mathcal{M} \models_{\nu,\eta} \varphi$ si $\nu^\sharp(t)$ et $\nu^\sharp(t')$ ont le même état déterministe ;*
- *si φ est de la forme* `after`$[t]\psi$, *alors $\mathcal{M} \models_{\nu,\eta} \psi$ si, pour tout couple $(v, \eta') \in M \times \underline{M}$ tel que $(\nu^\sharp(t), \eta)\mathrm{E}^\sharp(v, \eta')$, on a $\mathcal{M} \models_{\nu,\eta'} \psi$;*

On notera $\mathcal{M} \models_\Sigma \varphi$ si pour toute interprétation $\nu : \mathcal{X} \to M$ et tout $\eta \in \underline{M}$, on a $\mathcal{M} \models_{\nu,\eta} \varphi$.

2.1.2 κ-institution stratifiée

Le formalisme que nous venons de définir est présenté par la κ-institution stratifiée $\kappa\mathcal{SI} = (\mathcal{Sig}, \kappa, \mathcal{Mod}, [\![]\!])$ telle que :

- $\mathcal{Sig} = \mathcal{Sig}_*$ et $\mathcal{Mod} = \mathcal{Mod}_*$;
- $\kappa : \mathcal{Sig} \to \mathcal{LogSig}$ est un foncteur qui, à toute signature $\Sigma \in |\mathcal{Sig}|$, associe la signature de constructeurs $\kappa(\Sigma) = (\mathcal{C}^n)_{n \in \mathbb{N}} \in |\mathcal{LogSig}|$ définie de la façon suivante :
  - $\mathcal{C}^0 = \{t = t'/t, t' \in T_{\Sigma,s}(\mathcal{X}), s \in \mathcal{S}\} \cup \{t \equiv t'/t, t' \in T_\Sigma(\mathcal{X})\}$,
  - $\mathcal{C}^1 = \{\mathtt{after}[t]/t \in T_\Sigma(\mathcal{X})\}$;
  - $\forall n > 1$, $\mathcal{C}^n = \varnothing$.
- pour toute signature $\Sigma \in |\mathcal{Sig}|$ et pour tout Σ-modèle $\mathcal{M} \in |\mathcal{Mod}(\Sigma)|$, $[\![\mathcal{M}]\!]_\Sigma = \underline{M}$.

Il est clair que le formalisme que nous venons de définir tient à la fois de la logique équationnelle et de la logique modale munie de la modalité $\Box$. Il n'est cependant pas possible d'obtenir ce formalisme par combinaison de deux parchemins modèles-théoriques tel qu'indiqué en section 1.2 de cette troisième partie. Seulement, en y regardant de plus près, il est possible de séparer dans la syntaxe de ce formalisme ce qui provient de la logique équationnelle de ce qui vient de la logique modale. Pour toute signature $\Sigma \in |\mathcal{Sig}|$, on identifie alors deux sous-ensembles ${}^\Sigma\mathcal{C}_1$ et ${}^\Sigma\mathcal{C}_2$ de la signature de constructeurs $\kappa(\Sigma)$ associée à Σ tels que :

- ${}^\Sigma\mathcal{C}_1$ est la famille de constructeurs de formules définie de la façon suivante :
  - $\mathcal{C}_1^0 = \{t = t'/t, t' \in T_{\Sigma,s}(\mathcal{X}), s \in \mathcal{S}\} \cup \{t \equiv t'/t, t' \in T_\Sigma(\mathcal{X})\}$;
  - $\forall n > 0$, $\mathcal{C}_1^n = \varnothing$.
- ${}^\Sigma\mathcal{C}_2$ est la famille de constructeurs de formules définie de la façon suivante :
  - $\mathcal{C}_2^0 = \{t = t'/t, t' \in T_{\Sigma,s}(\mathcal{X}), s \in \mathcal{S}\} \cup \{t \equiv t'/t, t' \in T_\Sigma(\mathcal{X})\}$;
  - $\mathcal{C}_2^1 = \{\mathtt{after}[t]\}$;
  - $\forall n > 2$, $\mathcal{C}_2^n = \varnothing$.

Ceci définit alors naturellement deux foncteurs κ_1 et κ_2 tels que :

$$\begin{array}{rcl} \kappa_1 : \mathcal{Sig} & \to & \mathcal{LogSig} \\ \Sigma & \mapsto & {}^\Sigma\mathcal{C}_1 \end{array} \qquad \begin{array}{rcl} \kappa_2 : \mathcal{Sig} & \to & \mathcal{LogSig} \\ \Sigma & \mapsto & {}^\Sigma\mathcal{C}_2 \end{array}$$

On peut alors séparer la κ-institution stratifiée $\kappa\mathcal{SI}$ définie ci-dessus en deux sous-κ-institutions stratifiées $\kappa\mathcal{SI}_{|\kappa_1} = (\mathcal{Sig}, \kappa_1, \mathcal{Mod}, [\![]\!])$ et $\kappa\mathcal{SI}_{|\kappa_2} = (\mathcal{Sig}, \kappa_2, \mathcal{Mod}, [\![]\!])$. Ceci nous permet alors de définir deux morphismes $\mu_1 : \kappa SI_1 \to \kappa SI_{|\kappa_1}$ et $\mu_2 : \kappa SI_2 \to \kappa SI_{|\kappa_2}$, où $\kappa SI_{|\kappa_i}$ est la κ-institution stratifiée restreinte au foncteur κ_i.

Le morphisme $\mu_1 = (\Phi_1, \alpha_1, \beta_1, \Gamma_1) : \kappa SI_1 \to \kappa SI_{|\kappa_1}$ est défini de la façon suivante :

- $\Phi : \mathcal{Sig}_{\mathrm{EQ}} \to \mathcal{Sig}$ est le foncteur qui à toute signature équationnelle $\Sigma = (\mathcal{S}, \mathcal{F})$ associe la signature $\Phi_1(\Sigma) = (\mathcal{S} \cup \{\epsilon\}, \mathcal{F})$;

- $\alpha_1 : \kappa_1 \circ \Phi_1 \Rightarrow \kappa_{\mathrm{EQ}}$ est une transformation naturelle telle que pour toute signature $\Sigma \in |\mathit{Sig}|$, $\alpha_1(t = t') = t = t'$ et $\alpha_1(t \equiv t') = \mathtt{true}$;
- $\beta_1 : \mathit{Mod}_{\mathrm{EQ}} \Rightarrow \mathit{Mod} \circ \Phi_1^{op}$ est une tansformation naturelle telle que pour toute signature équationnelle $\Sigma \in |\mathit{Sig}_{\mathrm{EQ}}|$, $\beta_{1,\Sigma}$ associe à toute Σ-algèbre $\mathcal{A} = (A, \mathcal{F}^{\mathcal{A}}) \in |\mathit{Mod}_{\mathrm{EQ}}(\Sigma)|$ le $\Phi_1(\Sigma)$-modèle $(A, \underline{A}, \mathcal{F}^{\mathcal{A}}, \underline{\mathcal{F}^{\mathcal{A}}})$ tel que $\underline{A} = \{A'/(A', \mathcal{F}^{\mathcal{A}}) \simeq \mathcal{A}\}$ et $\underline{\mathcal{F}^{\mathcal{A}}} = \{f : A_{s_1} \times \cdots \times A_{s_n} \to \underline{A} / f \in \mathcal{F}_{s_1 \times \cdots \times s_n \to s}\}$;
- $\Gamma_1 = (\Gamma_{1,\Sigma'})_{\Sigma' \in |\mathit{Sig}_{\mathrm{EQ}}|}$ est une famille de transformations naturelles telles que pour toute signature $\Sigma' \in |\mathit{Sig}_{\mathrm{EQ}}|$ et toute Σ'-algèbre $\mathcal{A} \in |\mathit{Mod}_{\mathrm{EQ}}(\Sigma')|$, associe l'ensemble $\underline{A}$ à $[\![\mathcal{A}]\!]^{\mathrm{EQ}}_{\Sigma'}$.

Le morphisme $\mu_2 = (\Phi_1, \alpha_1, \beta_1, \Gamma_1) : \kappa\mathcal{S}I_2 \to \kappa\mathcal{S}I_{|\kappa_2}$ est défini de la façon suivante :

- $\Phi_2 : \mathit{Sig}_{\mathrm{MFOL}} \to \mathit{Sig}$ est le foncteur qi à toute signature $\Sigma \in |\mathit{Sig}_{\mathrm{MFOL}}|$ associe la signature $\Phi_2(\Sigma) = (\mathcal{S} \cup \{\epsilon\}, \mathcal{F})$;
- $\alpha_2 : \kappa_2 \circ \Phi_2 \Rightarrow \kappa_{\mathrm{MFOL}}$ est une transformation naturelle telle que pour toute signature $\Sigma \in |\mathit{Sig}|$, $\alpha_2(t = t') = t = t'$, $\alpha_1(t \equiv t') = t \equiv t'$ et $\alpha_2(\mathtt{after}[t]) = \Box_t$;
- $\beta_2 : \mathit{Mod}_{\mathrm{MFOL}} \Rightarrow \mathit{Mod} \circ \Phi_2^{op}$ est une transformation naturelle telle que pour toute signature $\Sigma \in |\mathit{Sig}_{\mathrm{MFOL}}|$, $\beta_{1,\Sigma}$ associe à tout Σ-modèle $\mathcal{M} = (M, W, \mathcal{R}_t) \in |\mathit{Mod}_{\mathrm{MFOL}}(\Sigma)|$ le quadruplet $(M, \underline{M}, \mathcal{F}^{\mathcal{M}}, \underline{\mathcal{F}^{\mathcal{M}}})$ tel que $\underline{M} = W$, $\mathcal{F}^{\mathcal{M}}$ est l'ensemble des symboles de fonctions rigides et $\underline{\mathcal{F}^{\mathcal{M}}}$ est l'ensemble des symboles de fonctions non-rigides de Σ ;
- $\Gamma_2 = (\Gamma_{2,\Sigma'})_{\Sigma' \in |\mathit{Sig}_{\mathrm{MFOL}}|}$ est une famille de transformations naturelles telles que pour toute signature $\Sigma' \in |\mathit{Sig}_{\mathrm{MFOL}}|$ et tout Σ'-modèle $\mathcal{M} \in |\mathit{Mod}_{\mathrm{MFOL}}(\Sigma')|$, associe l'ensemble $\underline{M}$ à $[\![\mathcal{M}]\!]^{\mathrm{EQ}}_{\Sigma'}$.

Via les modifications dénotées par les morphismes μ_1 et μ_2, la κ-institution stratifiée $\kappa\mathcal{SI}$ peut alors être vue comme une combinaison des κ-institutions stratifiées $\kappa\mathcal{SI}_1$ et $\kappa\mathcal{SI}_2$.

2.2 Définition de la combinaison de logiques

Nous définissons ici formellement l'approche que nous avons présentée à l'aide de notre exemple introductif dans la section précédente.

Définition 2.2.1 (combinaison) *Soit $(\kappa\mathcal{SI}_i)_{i \in I}$ une famille de κ-institutions stratifiées. Une κ-institution stratifée $\kappa\mathcal{SI}$ est une combinaison de $(\kappa\mathcal{SI}_i)_{i \in I}$ si :*

1. *il existe une famille de foncteurs $(\kappa_i : \mathit{Sig} \to \mathit{LogSig})_{i \in I}$ telle que pour toute signature $\Sigma \in |\mathit{Sig}|$, on a :*
   - $\bigcup_{i \in I} \kappa_i(\Sigma) = \kappa(\Sigma)$,
   - $\forall i \in I$, $\kappa(\Sigma)^0 \cap \kappa_i(\Sigma)^0 \neq \varnothing$.
2. *une famille de morphismes de κ-institutions stratifiées $(\mu_i : \kappa\mathcal{SI}_i \to \kappa\mathcal{SI}_{|\kappa_i})_{i \in I}$.*

Dans la suite, et lorsque cela n'engendre pas d'ambiguité, nous représenterons cette combinaison par la famille de foncteurs $(\kappa_i)_{i \in I}$.

Chapitre 3

Préservation de propriétés

Nous nous intéressons dans cette dernière section à la préservation de l'interpolation de Craig et du lemme de consistance de Robinson dans le cadre de la combinaison de logiques telle que nous l'avons définie dans la section précédente.
Soit donc $(\kappa_i)_{i \in I}$ une combinaison de logiques. Dans la partie I de cette thèse, nous avons déjà énoncé des conditions suffisantes assurant la préservation de l'interpolation de Craig et du lemme de consistance de Robinson au travers des morphismes d'institutions. Il nous reste alors à étudier la préservation de ces propriétés pour $\kappa\mathcal{SI}$ lorsque chacune des $\kappa\mathcal{SI}_{|\kappa_i}$ les vérifient. Nous abordons l'étude de la préservation de ces propriétés de deux façons :

- par *axiomatisation* des connecteurs ;
- par *préservation* de modèles.

3.1 Préservation par axiomatisation

L'idée est simple, nous allons supposer que chacun des connecteurs peut se spécifier par un ensemble de formules n'utilisant que des connecteurs apparaissant dans toutes les institutions stratifiées $\kappa\mathcal{SI}_{|\kappa_i}$ pour une combinaison de logiques $(\kappa_i)_{i \in I}$. Formellement, ceci s'exprime de la façon suivante.

Définition 3.1.1 *Soit $(\kappa_i)_{i \in I}$ une combinaison de logiques. Soit $\Sigma \in |\mathcal{S}ig|$ une signature. Alors, pour toute formule $\varphi \in T_{\kappa(\Sigma)} \setminus T_{\bigcap_{i \in I} \kappa_i(\Sigma)}$, il existe $E \subseteq T_{\bigcap_{i \in I} \kappa_i(\Sigma)}$ tel que $\mathcal{M}od(\varphi) = \mathcal{M}od(E)$.*

Cette condition est très proche de la condition de « conservativité pour les formules ». De la même façon, ceci suppose alors des contraintes sur chacune des institutions stratifiées $\kappa\mathcal{SI}_{|\kappa_i}$ comme par exemple de disposer de la négation (en effet, la négation ne se spécifie pas à partir de formules positives). Avec une telle conditions, on a alors le théorème suivant.

Théorème 3.1.1 *Soit $(\kappa_i)_{i \in I}$ une combinaison de logiques. Si chacune des $\kappa\mathcal{SI}_{\kappa_i}$ possède l'interpolation de Craig, alors $\kappa\mathcal{SI}$ également.*

Preuve À revoir ! Soit

$$\begin{array}{ccc} \Sigma & \xrightarrow{\sigma_1} & \Sigma_1 \\ {\scriptstyle\sigma_2}\downarrow & & \downarrow{\scriptstyle\sigma_1'} \\ \Sigma_2 & \xrightarrow[\sigma_2']{} & \Sigma' \end{array}$$

un diagramme dans $\mathcal{Sig}$ et soient $\Phi_1 \in \mathcal{Sen}(\Sigma_1)$ et $\Phi_2 \in \mathcal{Sen}(\Sigma_2)$ des ensembles de formules tels que $\Phi_1 \models_{\Sigma'} \Phi_2$. Puisque $\Phi_1 \models_{\Sigma'} \Phi_2$, on a $\mathcal{Mod}(\Phi_1) \subseteq \mathcal{Mod}(\Phi_2)$, *i.e.* $\mathcal{Mod}(\Phi_1 \cap \kappa(\Sigma')^0) \subseteq \mathcal{Mod}(\Phi_2 \cap \kappa(\Sigma')^0)$. Ceci implique alors $\Phi_1 \cap \kappa(\Sigma')^0 \models_{\Sigma'} \Phi_2 \cap \kappa(\Sigma')^0$. Par hypothèse, il existe alors un ensemble de formules atomiques $\Phi \in \kappa(\Sigma)^0$ interpolant de $\Phi_1 \cap \kappa(\Sigma')^0$ et $\Phi_2 \cap \kappa(\Sigma')^0$ et donc de Φ_1 et de Φ_2. □

3.2 Préservation par modèles

Nous donnons dans cette section une condition suffisante nous permettant de contourner le problème que certains connecteurs doivent nécessairement apparaitre nulle part ou alors dans chacune des κ-institutions stratifiées $\kappa\mathcal{SI}_{|\kappa_i}$ pour une combinaison de logiques $(\kappa_i)_{i\in I}$. Cette propriété va nous permettre de montrer non pas l'interpolation de Craig mais son équivalent (en présence de la négation) : la consistance de Robinson. Il suffira alors qu'une des κ-institutions stratifiées $\kappa\mathcal{SI}_{|\kappa_i}$ possède la négation et « le tour est joué ». Cette propriété, que nous appellerons $\mathcal{P}$ (à défaut d'un autre nom) s'exprime de la façon suivante.

Définition 3.2.1 *Soit $\kappa\mathcal{SI} = (\mathcal{Sig}, \mathcal{Mod}, [\![]\!], \models, \kappa)$ une κ-institution stratifiée. Soit $\Sigma \in |\mathcal{Sig}|$ et soit le diagramme ci-dessous*

$$\begin{array}{ccccc} & & T & & \\ & {\scriptstyle\sigma_1}\swarrow & & \searrow{\scriptstyle\sigma_2} & \\ T_1 & & & & T_2 \end{array}$$

où $T \subseteq \mathcal{Sen}(\Sigma)$ est une Σ-théorie complète et T_i ($i \in \{1,2\}$) est une σ_i-extension consistante de T. On dit que $\kappa\mathcal{SI}$ possède la propriété $\mathcal{P}$ si la propriété suivante est vérifiée :
$\forall \mathcal{M}_i \in |\mathcal{Mod}(T_i)|, \forall \mathcal{M}_j \in |\mathcal{Mod}(T_j)|, i \neq j \in \{1,2\}, \exists \mathcal{M}_k \in |\mathcal{Mod}(T_k)|, k \in \{1,2\}$

- $\mathcal{Mod}(\sigma_i)(\mathcal{M}_i) = \mathcal{Mod}(\sigma_j)(\mathcal{M}_k)$ *et* $\mathcal{M}_k <_\mu \mathcal{M}_j$ *si* $k = j$ *;*
- $\mathcal{Mod}(\sigma_j)(\mathcal{M}_j) = \mathcal{Mod}(\sigma_i)(\mathcal{M}_k)$ *et* $\mathcal{M}_k <_\mu \mathcal{M}_i$ *sinon (i.e.* $k = i$*).*

La propriété que nous venons de définir est une propriété assez forte qui, sous réserve de posséder la propriété d'amalgation (faible), entraine la propriété de Robinson et donc le lemme de consistance de Robinson. Cependant, cette propriété est vérifiée dans la pratique comme le montre le théorème suivant pour la logique du premier ordre.

Théorème 3.2.1 *La logique du premier ordre avec égalité (i.e. l'institution* FOL *introduite en partie I, exemple 1.1.3) verifie la propriété $\mathcal{P}$.*

$$\begin{tikzcd}
\Sigma \arrow[r, "\sigma_1"] \arrow[d, "\sigma_2"'] & \Sigma_1 \arrow[d, "\sigma'_1"] \\
\Sigma_2 \arrow[r, "\sigma'_2"'] & \Sigma'
\end{tikzcd}$$

FIG. 3.1 – *Pushout de signature dans* FOL

Preuve Soit le diagramme suivant de morphismes de signatures. Soit $T \subseteq \mathcal{Sen}(\Sigma)$ une Σ-théorie complète et soient $T_1 \subseteq \mathcal{Sen}(\Sigma_1)$ et $T_2 \subseteq \mathcal{Sen}(\Sigma_2)$ des Σ_i-extensions consistantes. On va montrer que l'ensemble de formules $\mathcal{Sen}(\sigma'_1)(T_1) \cup \mathcal{Sen}(\sigma'_2)(T_2)$ forme une Σ'-théorie consistante.
Soit donc $\mathcal{M}_1 \in \mathcal{Mod}(\Sigma_1)$ un modèle de T_1. Puisque cette dernière est consistante, l'existence de $\mathcal{M}_1$ est assurée par la proposition 1.3.6. Deux cas sont à considérer suivant que la cardinalité de $\mathcal{M}_1$ est ou non finie.

1. **$\mathcal{M}_1$ est un modèle fini :**
   Par définition, $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1)$ est aussi un modèle fini. Maintenant, par la condition de satisfaction $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1)$ est un modèle de T. Puisque T est une théorie complète, ceci signifie que la cardinalité de $\mathcal{M}_1$ est spécifiée dans T. Les modèles de T_2 sont alors tous des modèles finis dont la cardinalité est celle spécifiée dans T,*i.e.* celle de $\mathcal{M}_1$. Soit $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ un tel modèle. Soit $g : M_1 \to M_2$ la bijection entre les ensembles de valeurs sous-jacents de $\mathcal{M}_1$ et $\mathcal{M}_2$. Construisons alors un modèle $\mathcal{M}'_2$ élémentairement équivalent) $\mathcal{M}_2$ et tel que $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}'_2)$:
   - $M'_2 = M_1$,
   - $\forall f^n \in \mathcal{F}$, $f^{\mathcal{M}'_2} = f^{\mathcal{M}_1}$;
   - $\forall f^n \in \mathcal{F}_2 \setminus \mathcal{F}$,
   $$\begin{array}{rl} f^{\mathcal{M}'_2} : M_2^n & \to M_2 \\ (m_1, \ldots, m_n) & \mapsto f^{\mathcal{M}'_2}(g(m_1), \ldots, g(m_n)) \end{array}$$
   - $\forall r^n \in \mathcal{R}$, $r^{\mathcal{M}'_2} = r^{\mathcal{M}_1}$;
   - $\forall r^n \in \mathcal{R}_2 \setminus \mathcal{R}$, $r^{\mathcal{M}'_2} = \{(m_1, \ldots, m_n)/(g(m_1), \ldots, g(m_n)) \in r^{\mathcal{M}_2}\}$
2. **$\mathcal{M}_1$ est un modèle de cardinalité $\alpha \geq \aleph_0$:**
   De la même façon que précédemment $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1)$ est un modèle de cardinalité α. La théorie T étant complète, il s'ensuit que les modèles de T_2 sont également de cardinalité infinie. Par le théorème de Löwenheim-Skolem, T_2 admet un modèle pour toute cardinalité infinie. Il existe donc $\mathcal{M}_2 \in |\mathcal{Mod}(\Sigma_2)|$ de cardinalité α. En suivant la démarche ci-dessus, on peut alors construire un modèle $\mathcal{M}'_2$ de T_2 élémentairement équivalent à $\mathcal{M}_2$ et tel que $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}'_2)$.

Par construction, on a bien $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}'_2)$. Par définition, l'application $g : \mathcal{M}'_2 \to \mathcal{M}_2$ est un isomorphisme. On en déduit alors que $\mathcal{M}'_2 \equiv \mathcal{M}_2$. Enfin, par construction on a bien $\mathcal{Mod}(\sigma_1)(\mathcal{M}_1) = \mathcal{Mod}(\sigma_2)(\mathcal{M}'_2)$. □

Comme on peut le voir sur cet exemple si une κ-institution stratifiée donnée préserve la satisfaction, possède la négation et des diagrammes complets et vérifie les théorèmes de Löwenheim-Skolem, alors elle possède la propriété $\mathcal{P}$. Il est possible que la propriété $\mathcal{P}$ dépende d'hypothèses moins restrictive que celles-ci (de fait cette construction est très liée aux structures du premier ordre et l'hypothèses des diagrammes complets exclu les logiques modales) que nous n'avons pas encore identifiées.

Théorème 3.2.2 *Pour toute institution qui satisfait la propriété $\mathcal{P}$ et possède la propriété de faible amalgation le lemme de consistance de Robinson est vrai.*

Preuve Soit le diagramme suivant

$$\begin{tikzcd} & T \arrow[dl, "\sigma_1"'] \arrow[dr, "\sigma_2"] & \\ T_1 & & T_2 \end{tikzcd}$$

où $T \subseteq \mathcal{Sen}(\Sigma)$ est une Σ-théorie complète et T_i ($i \in \{1,2\}$) est une σ_i-extension consistante de T. Par la propriété $\mathcal{P}$, on a :
$\forall \mathcal{M}_i \in |\mathcal{Mod}(T_i)|, \forall \mathcal{M}_j \in |\mathcal{Mod}(T_j)|, i \neq j \in \{1,2\}, \exists \mathcal{M}_k \in |\mathcal{Mod}(T_k)|, k \in \{1,2\}$

- $\mathcal{Mod}(\sigma_i)(\mathcal{M}_i) = \mathcal{Mod}(\sigma_j)(\mathcal{M}_k)$ et $\mathcal{M}_k <_\mu \mathcal{M}_j$ si $k = j$;
- $\mathcal{Mod}(\sigma_j)(\mathcal{M}_j) = \mathcal{Mod}(\sigma_i)(\mathcal{M}_k)$ et $\mathcal{M}_k <_\mu \mathcal{M}_i$ sinon (*i.e.* $k = i$).

Par la propriété d'amalgation faible, il existe alors un Σ'-modèle $\mathcal{M}' \in |\mathcal{Mod}(\Sigma)|$ tel que $\mathcal{Mod}(\sigma_k)(\mathcal{M}') = \mathcal{M}_k$ pour $k \in \{1,2\}$. Par la condition de satisfaction, on en déduit que $\mathcal{M}' \models_{\Sigma'} \mathcal{Sen}(\sigma'_1)(T_1) \cup \mathcal{Sen}(\sigma'_2)(T_2)$ et donc $\mathcal{Sen}(\sigma'_1)(T_1) \cup \mathcal{Sen}(\sigma'_2)(T_2)$ est une Σ'-théorie consistante. □

La théorème précédent établit une nouvelle preuve du lemme de consistance de Robinson plus simple que les preuves existantes de ce lemme. Ceci est du au fait que toute la difficulté de montrer ce lemme est « cachée » dans la propriété $\mathcal{P}$.

Corollaire 3.2.1 *Toute institution qui satisfait la propriété $\mathcal{P}$ et possède la propriété de faible amalgation et la négation possède l'interpolation de Craig.*

Preuve C'est une conséquence immédiate de la proposition ci-dessus et de la proposition 2.2.2, partie I. □

On montre alors que cette propriété est préservée par combinaison de logiques au sens de la définition 2.2.1

Théorème 3.2.3 *Soit $(\kappa_i)_{i \in I}$ une combinaison de logiques. Si chacune des $\kappa\mathcal{SI}_{|\kappa_i}$ possède la propriété $\mathcal{P}$, alors $\kappa\mathcal{SI}$ également.*

Preuve Soit $\kappa\mathcal{SI} = (\mathcal{Sig}, \mathcal{Mod}, [\![]\!], \models, \kappa)$ une κ-institution stratifiée. Soit $\Sigma \in |\mathcal{Sig}|$ et soit le diagramme ci-dessous

$$\begin{tikzcd} & T \arrow[dl, "\sigma_1"'] \arrow[dr, "\sigma_2"] & \\ T_1 & & T_2 \end{tikzcd}$$

où $T \subseteq \mathcal{S}en(\Sigma)$ est une Σ-théorie complète et T_i ($i \in \{1,2\}$) est une σ_i-extension consistante de T. Soient $\mathcal{M}_1 \in \mathcal{M}od(T_1)$ et $\mathcal{M}_2 \in \mathcal{M}od(T_2)$ des modèles. Puisque $\kappa\mathcal{SI}_{|\kappa_1}$ et $\kappa\mathcal{SI}_{|\kappa_2}$ possèdent la propriété $\mathcal{P}$, on a pu construire un T_2-modèle $\mathcal{M}$ dont le réduit par σ_2 est égale au réduit de $\mathcal{M}_1$ par σ_1 et qui est élémentairement équivalent à $\mathcal{M}_2$ à la fois lorsqu'on se restreint aux constructeurs de formules donnés par κ_1 et à ceux donnés par κ_2. Il faut maintenant montrer que $\mathcal{M}$ est toujours équivalent à $\mathcal{M}_2$ lorsqu'on considère tous les constructeurs de formules données par κ. Supposons donc le contraire. Ceci signifie qu'il existe un formule $\varphi \in T_2$ telle que soit $\mathcal{M} \models \varphi$ et $\mathcal{M}_2 \not\models \varphi$ soit $\mathcal{M} \not\models \varphi$ et $\mathcal{M}_2 \models \varphi$. Par hypothèse, cette formule est nécessairement une formule mixte, *i.e.* construite sur les deux langages. Ce n'est donc pas une formule atomique et elle est donc de la forme $\top(\varphi_1, \ldots, \varphi_n)$. Supposons que $\top$ soit un constructeur de formules donné par κ_1. Alors, φ est équivalente à la formule $\top(b_1, \ldots, b_n)$ où les b_i sont les valeurs de vérité des formules φ_i. Or, $\top(b_1, \ldots, b_n)$ est une formule construite sur les constructeurs de formules donnés par κ_1. Ceci est imposible puisque $\mathcal{M}$ et $\mathcal{M}_2$ sont élémentairement équivalents sur l'ensemble des formules construites sur les constructeurs donnés par κ_1.
On répète alors ce qu'on vient de faire pour $\top$ donné par κ_2 et pour $\mathcal{M}$ T_1-modèle élémentairement équivalent à $\mathcal{M}_1$. $\square$

Corollaire 3.2.2 *Soit $(\kappa_i)_{i \in I}$ une combinaison de logiques. Si chacune des $\kappa\mathcal{SI}_{|\kappa_i}$ possède la propriété $\mathcal{P}$, alors $\kappa\mathcal{SI}$ possède l'interpolation de Craig.*

Preuve C'est une conséquence directe de la proposition 3.2.3 et du corollaire 3.2.1. $\square$

Conclusion

Apports de la thése

L'apport de cette thèse à la théorie générale des institutions et des systèmes logiques est double. Cette dualité se retrouve d'ailleurs dans le découpage des parties. En effet, tandis que les résultats présentés dans la troisième partie et, dans une moindre mesure dans la première, concernent essentiellement la théorie de la spécification, ceux présentés dans la deuxième partie relève plutot de la théorie des modèles.
En ce qui concerne la théorie de la spécification, les apports de cette thèse sont les suivants :

1. Généralisation de l'équivalence entre l'interpolation de Craig et la consistance de Robinson (*cf.* partie I, section 2.2.2). Le résultat que nous proposons s'inspire et améliore sensiblement celui proposé par A. Salibra et G. Scollo (*cf.* [114]).
2. Préservation de l'interpolation de Craig au-travers des morphismes d'institutions (*cf.* partie I, section 2.3.2). Là encore, le résultat que nous proposons s'inspire d'un résultat analogue démontré par A. Salibra et G. Scollo pour les transformation de (pré)-institutions.
3. Introduction d'un cadre théorique basé sur les institutions, appelé institutions stratifiées, permettant de préciser, de facon institution-indépendante, la définition de la relation de satisfaction au-travers des éléments utilisés pour définir la validité des formules (*cf.* partie II, chapitre 1).
4. Caractérisation de la notion de constructeurs de formules et de leurs sémantiques dans le cadre des institutions stratifiées par la donnée d'un foncteur associant à toute signature une signature de constructeurs munie d'une algèbre pour la sémantique (*cf.* partie II, section 3.1).
5. Utilisation de ce dernier cadre théorique, appelé κ-institutions stratifiées, pour létude de la combinaison de logiques incluant la notion de modifications des logiques combinées (*cf.* partie III, chapitre 2) et la prérservation de línterpolation de Craig et de la consistance de Robinson au travers de cette combinaison (*cf.* partie III, section 3).

En ce qui concerne la théorie des modéles, l'apport principal de cette thèse est la généralisation de la méthode des diagrammes (*cf.* partie II, section 2.2), rendue possible par l'introduction du cadre des institutions stratifiées. La méthode des diagrammes est en effet à la base de nombreux résultats de théorie standard des modèles (comme par exemple les théorèmes de Löwenheim-Skolem ou le test de Tarski-Vaught) et sa généralisation dans un cadre institution-indépendant ouvre la voie à la généralisation de ces résultats. En particulier, cette généralisation de la méthode des diagrammes nous a permi de donner des conditions suffisantes pour la consistance de Robinson (*cf.* partie II, section 3.3), et donc pour l'interpolation de Craig, ce qui, hormis de très récents travaux par D. Găină et A. Popescu, n'a jamais été fait malgré le role joué par ces propriétés dans le cadre de la structuration et de la modularité des spécifications axiomatiques. Les autres apports de cette thèse à la théorie des modèles sont les suivants :

1. preuves institutions-indépendantes des théorème de Löwenheim-Skolem, du test de Tarski-Vaught et du théorème des chaines élémentaires de Tarski (*cf.* partieII, sections 2.3 et 3.2) en application directe de la méthode des diagrammes.
2. formulation dans un cadre institution-indépendant et généralisation de la notion de définissabilité de Beth (*cf.* partie I, sections 3.2 et 3.5).
3. preuve institution-indépendante et préservation du théorème de définissabilité de Beth (*cf.* partie I, sections 3.3 et 3.4) en application directe du théorème d'interpolation de Craig.

Perspectives

Comme on peut le deviner à la lecture de la section précédente, les travaux ici présentés laissent apparaitre de nombreuses perspectives, à la fois dans le cadre de la théorie de la spécification et de la théorie des modèles. On retiendra notamment les perspectives de recherches suivantes :

Théorie de la spécification :

1. Approfondir l'étude la combinaison de logiques dans le cadre des κ-institutions stratifiées, par exemple em considérant d'autres notions de flèches entre système logiques (co-morphismes, transformations, *etc.*).
2. Approfondir l'étude de la notion de stratification (pouvoir, limites, *etc.*). En particulier, il serait intéressant d'associer à chaque formule l'ensemble des états d'un modèle donné pour lesquels cette formule est vraie. Ceci permettrait de voir une formule comme un simple opérateur de fermeture sur l'ensemble des états associés à chaque modèle et ainsi de mieux caractériser la sémantique des constructeurs de formules.
3. Étudier la préservation d'autres propriétés utiles pour la théorie de la spécification (décidabilité, axiomatisabilité, correction, complétude, *etc.*) au-travers de notre définition de la combinaison de logiques.

Théorie des modèles :

1. Exploration de la théorie standard des modèles dans le cadre des institutions stratifiées (généralisation et préservation d'autres résultats).
2. comparaison des institutions stratifiées avec d'autres cadres théoriques tels que les power-institutions (*cf.* [86]).
3. comparaison des pouvoirs d'expression relatifs de divers systèmes logiques dans le cadre des institutions stratifiées au-travers de leurs classes de modèles.

Quatrième partie

Annexes

Annexe A

Théorie des catégories

A.1 Catégories, foncteurs et transformations naturelles

Définition A.1.1 (graphe) *Un* graphe $\mathcal{G}$ *est la donnée d'un ensemble* $\mathcal{O}$ *d'objets, d'un ensemble* $\mathcal{F} \subseteq \mathcal{O} \times \mathcal{O}$ *de flèches et de deux applications* $\mathtt{dom} : \mathcal{F} \to \mathcal{O}$ *et* $\mathtt{codom} : \mathcal{F} \to \mathcal{O}$ *associant respectivement à toute flèche* $f \in \mathcal{F}$ *son domaine et son codomaine, i.e.* $f : \mathtt{dom}(f) \to \mathtt{codom}(f)$.

Définition A.1.2 (paire composable) *Soit* $\mathcal{G} = (\mathcal{O}, \mathcal{F})$ *un graphe. L'ensemble des paires composables de* $\mathcal{G}$*, noté* $\mathcal{F} \times_{\mathcal{O}} \mathcal{F}$*, est défini de la façon suivante :*

$$\mathcal{F} \times_{\mathcal{O}} \mathcal{F} = \{(g, f) \in \mathcal{F} \times \mathcal{F} / \mathtt{dom}(g) = \mathtt{codom}(f)\}$$

L'ensemble des paires composables est appelé ensemble produit sur $\mathcal{O}$.

Définition A.1.3 (catégorie) *Une catégorie* $\mathcal{C}$ *est la donnée d'un graphe* $\mathcal{G} = (\mathcal{O}, \mathcal{F})$ *et des deux applications suivantes :*

- $\mathtt{Id} : \begin{array}{ll} \mathcal{O} & \to \mathcal{F} \\ c & \mapsto \mathtt{Id_c} \end{array}$
- $\circ : \begin{array}{ll} \mathcal{F} \times_{\mathcal{O}} \mathcal{F} & \to \mathcal{F} \\ (g, f) & \mapsto g \circ f \end{array}$

vérifiant les propriétés suivantes :

1. $\mathtt{dom}(\mathtt{Id_a}) = \mathtt{a} = \mathtt{codom}(\mathtt{Id_a})$;
2. $\mathtt{dom}(g \circ f) = \mathtt{dom} f$;
3. $\mathtt{codom}(g \circ f) = \mathtt{codom}(g)$;
4. $k \circ (g \circ f) = (k \circ g) \circ f$;
5. $\forall f : a \to b$, $\mathtt{Id_b} \circ \mathtt{f} = \mathtt{f}$;
6. $\forall g : b \to c, g \circ \mathtt{Id_b} = \mathtt{g}$.

Notation A.1.1 *Soit* $\mathcal{C}$ *une catégorie.*

- *On note* $|\mathcal{C}|$ *la classe des objets de* $\mathcal{C}$.
- *Pour tout couple* $(a, b) \in |\mathcal{C}| \times |\mathcal{C}|$ *d'objets de* $\mathcal{C}$*, on note* $\mathcal{C}(a, b)$ *la classe des flèches de* a *vers* b*. Une flèche* $f \in \mathcal{C}(a, b)$ *sera désormais appelée* morphisme de a vers b.

Une catégorie pouvant être définie uniquement en termes de ses morphismes, on ne fera pas de distinction dans les notations entre la catégorie $\mathcal{C}$ et l'ensemble de tous ses morphismes.

On définit maintenant la notion de foncteur. Intuitivement, un foncteur est un morphisme de catégories.

Définition A.1.4 (foncteur) *Soient $\mathcal{C}$ et $\mathcal{B}$ deux catégories. Un* foncteur $T : \mathcal{C} \to \mathcal{B}$ *est la donnée d'un couple $(T_O, T_{\mathcal{F}})$ d'applications tel que :*

- $T_O : |\mathcal{C}| \to |\mathcal{B}|$, *appelée* fonction objet, *associe à tout objet $c \in |\mathcal{C}|$ un objet $T(c) \in |\mathcal{B}|$;*
- $T_{\mathcal{F}} : \mathcal{C} \to \mathcal{B}$, *appelée* fonction flèche, *associe à tout morphisme $f \in \mathcal{C}(c, c')$ une flèche $T(f) \in \mathcal{B}(T(c), T(c'))$;*

vérifiant les deux propriétés suivantes :

1. $T(\mathtt{Id_c}) = \mathtt{Id_{T(c)}}$;
2. $T(g \circ f) = T(g) \circ T(f)$ *(si $g \circ f$ est défini dans $\mathcal{C}$).*

Suivant la même démarche que précédemment, on définit maintenant la notion de transformation naturelle. Intuitivement, une transformation naturelle est un morphisme de foncteur.

Définition A.1.5 (transformation naturelle) *Soient $S, T : \mathcal{C} \to \mathcal{B}$ deux foncteurs. Une* transformation naturelle $\tau : S \Rightarrow T$ *est une application qui associe à tout objet $c \in \mathcal{C}$ une flèche $\tau_c \in \mathcal{B}(S(c), T(c))$ de telle façon que pour tout morphisme $f \in \mathcal{C}(c, c')$ le diagramme de la figure suivante commute. On*

$$\begin{tikzcd} c \arrow[d, "f"'] & S(c) \arrow[r, "\tau_c"] \arrow[d, "S(f)"'] & T(c) \arrow[d, "T(f)"] \\ c' & S(c') \arrow[r, "\tau_{c'}"'] & T(c') \end{tikzcd}$$

dit alors que $\tau_c : S(c) \to T(c)$ est naturel *en c.*

Si on imagine que les foncteurs S et T définissent chacun une certaine image de $\mathcal{C}$ dans $\mathcal{B}$, alors une transformation naturelle τ est un ensemble de flèches associant à l'image définie par S, celle définie par T.

A.2 Fondations

L'un des objectifs principaux de la théorie des catégories est de discuter de propriétés de totalités d'objets mathématiques tels que l'« ensemble » de tous les groupes ou l'« ensemble » de tous les homomorphismes de groupes. Or, il est usuel désormais de voir les structures algébriques telles que les groupes comme des ensembles munis d'une structure particulière. On va donc chercher à considérer l'ensemble de tous les ensembles possédant une structure particulière donnée. Ceci nous amène alors à appliquer un principe de compréhension pour former notre ensemble : étant donnée une propriété φ portant sur des ensembles, on forme l'ensemble $\{x/\varphi(x)\}$ de tous les ensembles x vérifiant la propriété φ (dénoté $\varphi(x)$). Un tel principe, cependant, ne peut être adopté dans sa généralité car il mènerait à certains paradoxes fameux tels que le paradoxe de Russel. Pour cette raison, la pratique standard en théorie des ensembles naïve (*i.e.* avec la relation d'appartenance classique $\in$) est de restreindre l'application de ce principe de compréhension. On n'autorise plus, à partir de deux ensembles u et v, que les constructions ensemblistes suivantes :

- $\{u, v\}$ l'ensemble formé exclusivement des éléments u et v ;
- (u, v) la paire ordonnée ;
- $u \times v = \{(x, y)/x \in u \text{ et } y \in v\}$ le produit cartésien de u par v;
- $\wp(u) = \{v/v \subset u\}$ l'ensemble des parties de u ;
- $\bigcup x = \{y/y \in z \text{ pour un certain } z \in x\}$ l'union (d'une chaîne x d'ensembles) ;
- d'un ensemble infini de tous les ordinaux ($\omega = \{0, 1, 2, \ldots\}$).

Finalement, étant donnée une propriété φ portant sur des ensembles (techniquement une propriété exprimée en termes d'éléments, de la relation d'appartenance et des connecteurs logiques et des quantificateurs usuels) et étant donné un ensemble u, on autorise alors le principe de compréhension pour les éléments de u, *i.e.* l'ensemble $\{x \in u/\varphi(x)\}$. Ceci signifie qu'en fait on autorise la formation de l'ensemble de tous les ensembles appartenant à un ensemble donné u et vérifiant une propriété donnée. On ajoute alors à cette pratique une assomption supplémentaire : l'existence d'un univers.

Définition A.2.1 (univers) *Un* univers *est la donnée d'un ensemble U vérifiant les conditions suivantes :*

- $x \in u \in U$ *implique* $x \in U$;
- $u, v \in U$ *implique* $\{u, v\}, (u, v), u \times v \in U$;
- $x \in U$ *implique* $\wp(x), \bigcup x \in U$;
- $\omega \in U$ *(ici, nous considèrerons* $\omega = \{0, 1, 2, \ldots\}$*) est l'ensemble de tous les ordinaux ;*
- *si* $f : a \to b$ *est une fonction sujective avec* $a \in U$ *et* $b \subset U$*, alors* $b \in U$.

Ces propriétés de fermeture sur U assurent que les opérations usuelles de la théorie des ensembles appliquées aux éléments de U produiront toujours des éléments de U.

Définition A.2.2 (petit ensemble) *Soit U un univers. On appelle* petit ensemble *tout élément de U, i.e. tout ensemble $u \in U$.*

Remarque A.2.1 *Soit U un univers. L'ensemble $\{U\}$ est un ensemble ne contenant qu'un seul élément : l'univers U. Intuitivement c'est donc un ensemble très petit. Il n'est pourtant pas petit au sens de la définition précédente. En effet, $\{U\} \in U$ impliquerait $U \in U$ ce qui est en contradicion avec l'axiome de régularité (aucun ensemble n'est élément de lui-même). Par conséquent, « petit » ensemble signifie ensemble élément d'un univers et non ensemble de faible cardinalité.*

D'après la définition A.2.2, l'univers U est alors l'ensemble de tous les petits ensembles. Une fonction étant un ensemble (le triplet (u, G_f, v), où $G_f \subset u \times v$), on dira qu'une fonction $f : u \to v$ est petite si u et v sont de petits ensembles.

On peut donc construire l'ensemble A de tous les ensembles qui sont des petites fonctions et ainsi définir la catégorie des petits ensembles dont les morphismes sont les petites fonctions.

Notation A.2.1 *Soit U un univers. On note $\mathcal{Set}$ la catégorie dont les objets sont les petits ensembles et les morphismes les petites fonctions, i.e. :*
- $|\mathcal{Set}| = U$;
- $\mathcal{Set} = \{f : a \to b / a, b \in U\}$.

Définition A.2.3 (petite catégorie) *Soit U un univers. Une catégorie $\mathcal{C}$ est dite* petite *si l'ensemble de ses objets et l'ensemble de ses flèches sont tous deux de petits ensembles, i.e. :*
- $|\mathcal{C}| \in U$;
- $\mathcal{C} \in U$.

Notation A.2.2 *Soit U un univers. On note $\mathcal{Cat}$ la catégorie dont l'ensemble des objets est l'ensemble des petites catégories et l'ensemble des morphismes est l'ensemble des petits foncteurs, i.e. les foncteurs entre petites catégories.*

Remarque A.2.2 *On remarquera que $\mathcal{Set}$ n'est pas une petite catégorie puisque l'ensemble U de ses objets n'est pas petit ($U \notin U$).*

Dans les applications de la théorie des catégories à la théorie de la spécification, il est fréquent de rencontrer des catégories non-petites. D'ailleurs, la plupart des institutions standards ont, pour une signature donnée, un ensemble de modèles qui n'est pas petit. Il est alors commode d'utiliser les axiomes de Gödel-Bernays pour la théorie des ensembles et définir la notion de classe.

Définition A.2.4 (classe) *Soit U un univers. Une* classe *C est la donnée d'un sous-ensemble de U, i.e. $C \subseteq U$.*

On remarque immédiatement que, puisque $x \in u \in U$ implique $x \in U$, tout élément de U est également un sous-ensemble de U. On en déduit que tout petit ensemble de U est une classe. Par contre, certaines classes de U, tel que U lui-même, ne sont pas des petits ensembles.

Définition A.2.5 (classe propre) *Soit U un univers. Une* classe propre *est la donnée d'une classe C qui n'est pas élément de U, i.e. $C \subseteq U$ et $C \notin U$.*

On peut maintenant définir la notion de large catégorie.

Définition A.2.6 (large catégorie) *Soit U un univers. Une catégorie $\mathcal{C}$ est dite* large *si l'ensemble de ses objets et l'ensemble de ses flèches sont tous deux des classes (propres ou non), i.e. :*
- $|\mathcal{C}| \subseteq U$;
- $\mathcal{C} \subseteq U$.

Remarque A.2.3 *Les catégories $\mathcal{Set}$ et $\mathcal{Cat}$ sont des catégories larges.*

Dans l'ensemble de ce document, nous n'indiquerons pas l'univers U choisi et nous parlerons de petits ensembles, de classes et d'ensembles, sachant que le terme « ensemble » regroupe les notions de petit ensemble et de classe.

A.3 Constructions sur les catégories

Contravariance et dualité

Définition A.3.1 (catégorie duale) *Soit $\mathcal{C}$ une catégorie. On définit la* catégorie duale *de $\mathcal{C}$, notée $\mathcal{C}^{op}$, de la façon suivante :*

- *les objets de $\mathcal{C}^{op}$ sont les objets de $\mathcal{C}$;*
- *les objets de $\mathcal{C}^{op}$ sont les morphismes f^{op} obtenus à partir des morphismes de $\mathcal{C}$ en inversant les directions, i.e. :*
  $\forall a, b \in |\mathcal{C}|$,
$$\mathcal{C}^{op}(b, a) = \{f^{op} : b \to a / \exists! f \in \mathcal{C}(a, b)\}$$

Définition A.3.2 (contravariant) *Un* foncteur contravariant $S : \mathcal{C} \to \mathcal{B}$ *est la donnée d'un foncteur $\overline{S} : \mathcal{C}^{op} \to \mathcal{B}$, i.e. à tout objet $c \in \mathcal{C}$, S associe un objet $S(c) \in |\mathcal{B}|$ et à tout morphisme $f \in \mathcal{C}(a, b)$ S associe un morphisme $S(f) \in \mathcal{B}(S(b), S(a))$ tel que les deux conditions suivantes soient vérifiées :*

1. $S(\mathtt{Id_c}) = \mathtt{Id_{S(c)}}$
2. $S(g \circ f) = S(f) \circ S(g)$

La généralisation des notions d'injectivité et de surjectivité dans le langage catégoriel offre un bel exemple de notions duales.
Dans le cadre de l'algèbre abstraite ou de l'algèbre universelle, un monomorphisme est simplement la donnée d'un morphisme injectif.

Définition A.3.3 (monomorphisme) *Soit $\mathcal{C}$ une catégorie et soit $f \in \mathcal{C}(a, b)$. Le morphisme f est un* monomorphisme *si pour tous morphismes $g_1, g_2 \in \mathcal{C}(c, a)$, on a :*
$$f \circ g_1 = f \circ g_2 \Rightarrow g_1 = g_2$$

Dans la catégorie $\mathcal{S}et$ des ensembles, les monomorphismes sont exactement les morphismes injectifs. Les notions algébriques et catégorielles coïncident donc. Ceci est également vrai pour de nombreuses autres catégories telles que la catégorie des groupes, celles des anneaux ou celle des espaces vectoriels. En revanche, il existe des catégories, telles que la catégorie des groupes abéliens divisibles munis des homomorphismes de groupes, pour lesquelles il existe des monomorphismes non injectifs.
Dans le cadre de l'algèbre abstraite ou de l'algèbre universelle, un épimorphisme est simplement la donnée d'un morphisme surjectif.

Définition A.3.4 (épimorphisme) *Soit $\mathcal{C}$ une catégorie et soit $f \in \mathcal{C}(a, b)$. Le morphisme f est un* épimorphisme *si pour tous morphismes $g_1, g_2 \in \mathcal{C}(b, c)$, on a :*
$$g_1 \circ f = g_2 \circ f \Rightarrow g_1 = g_2$$

Dans la catégorie $\mathcal{S}et$ des ensembles, les épimorphismes sont exactement les morphismes surjectifs. Les notions algébriques et catégorielles coincident donc. Cependant, de même que pour les monomorphismes, il existe des catégories dont certains épimorphismes ne sont pas surjectifs. Par exemple :

- dans la catégorie des monoïdes, l'inclusion de $\mathbb{N}$ dans $\mathbb{Z}$ est un homomorphisme de monoïdes non-surjectif. C'est pourtant un épimorphisme ;
- dans la catégorie des anneaux, l'inclusion de $\mathbb{Z}$ dans $\mathbb{Q}$ est un épimorphisme non-surjectif.

Il est clair d'après les définitions que le dual d'un monomorphisme est un épimorphisme, *i.e.* un monomorphisme dans une catégorie $\mathcal{C}$ est un épimorphisme dans la catégorie duale C^{op}.

Définition A.3.5 (isomorphisme) *Soit $\mathcal{C}$ une catégorie et soit $f \in \mathcal{C}(a,b)$. Le morphisme f est un* isomorphisme *si c'est un monomorphisme et un épimorphisme.*

Définition A.3.6 (monomorphisme extrême) *Soit $\mathcal{C}$ une catégorie et soit $f \in \mathcal{C}(a,b)$ un monomorphisme. On dit que f est un* monomorphisme extrême *si pour toute factorisation de f, le premier facteur n'est pas un épimorphisme, ou alors c'est un isomorphisme, i.e. :*
$\forall c \in |\mathcal{C}|, \forall g_1 \in \mathcal{C}(a,c), \forall g_2 \in \mathcal{C}(c,b),$

$$f = g_1 \circ g_2 \wedge g_1 \textit{ épimorphisme } \Rightarrow g_1 \textit{ isomorphisme}$$

Catégories virgules

Définition A.3.7 (catégorie virgule) *Soit $F : \mathcal{C}' \to \mathcal{C}$ un foncteur et $a \in |\mathcal{C}|$ un objet de la catégorie $\mathcal{C}$. La* catégorie virgule a/F *est définie de la façon suivante :*

objets : *les morphismes $f \in \mathcal{C}(a, F(b))$, où $b \in |\mathcal{C}'|$;*

morphismes : *les morphismes $h \in \mathcal{C}'(b,b')$, tels que $F(h) \circ f = f'$ si $h \in a/F(f,f')$.*

$$\begin{tikzcd}
a \arrow[r, "f"] \arrow[dr, "f'"'] & F(b) \arrow[d, "F(h)"] \\
& F(b')
\end{tikzcd}$$

Notation A.3.1 *Lorsque $\mathcal{C} = \mathcal{C}'$ et que F est le foncteur identité, la catégorie a/F sera notée $a/\mathcal{C}$.*

A.4 Universalité et limites

Flèches universelles

Définition A.4.1 (flèche universelle) *Si $S : \mathcal{D} \to \mathcal{C}$ est un foncteur et $c \in |\mathcal{C}|$ est un objet de $\mathcal{C}$, une* flèche universelle de c vers S *est la donnée d'une paire (r,u), où $r \in |\mathcal{D}|$ est un objet de $\mathcal{D}$ et $u \in \mathcal{C}(c, S(r))$ est une flèche de $\mathcal{C}$, telle que pour toute autre paire (d,f) ($d \in |\mathcal{D}|$ et $f \in \mathcal{C}(c, S(d))$), il existe une unique flèche $f' \in \mathcal{D}(r,d)$ de $\mathcal{D}$ telle que $S(f') \circ u = f$.*

La définition précédente signifie en fait que chaque flèche f vers S se factorise de façon unique au travers de la flèche universelle u comme indiqué sur le diagramme ci-dessous.

$$\begin{tikzcd}
c \arrow[r, "u"] \arrow[dr, "f"'] & S(r) \arrow[d, "S(f')"] & r \arrow[d, "f'"] \\
& S(d) & d
\end{tikzcd}$$

De façon équivalente, une flèche $u : c \to S(r)$ est universelle de c dans S lorsque la paire (r, u) est un objet initial de la catégorie virgule (c/S) dont les objets sont les flèches $c \to S(d)$. Comme toujours avec les objets initiaux, cela signifie que (r, u) est unique à isomorphisme près dans (c/S). En particulier, l'objet $r \in |\mathcal{D}|$ est unique à isomorphisme près dans $\mathcal{D}$. Cette remarque est typique de l'emploi des catégories virgules.

L'idée d'universalité est parfois exprimée en termes d'« éléments universels ». Si $\mathcal{D}$ est une catégorie et $H : \mathcal{D} \to \mathcal{S}et$ un foncteur, un *élément universel* du foncteur H est la donnée d'une paire (r, e), où $r \in |\mathcal{D}|$ est un objet de D et $e \in H(r)$ est un élément de l'ensemble associé à r par H, telle que pour toute paire (d, x) ($d \in |\mathcal{D}|$ et $x \in H(d)$), il existe une unique flèche $f \in \mathcal{D}(r, d)$ telle que $H(f)(e) = x$.
Beaucoup de constructions familiaires sont des exemples naturels d'éléments universels. Considérons par exemple une relation d'équivalence E sur un ensemble S, l'ensemble quotient S/E des classes d'équivalences et la projection $p : S \to S/E$ qui associe à tout élément $s \in S$ sa classe d'équivalence. L'ensemble quotient S/E possède la propriété suivante : toute fonction f sur S respectant la relation d'équivalence peut être vue comme une fonction sur S/E. Plus formellement, ceci signifie que si $f : S \to X$ est telle que $f(s) = f(s')$ pour tout $s, s' \in S$ tels que sEs', alors f peut être écrite de façon unique comme la composée p avec une fonction $f' : S/E \to X$.

$$\begin{tikzcd} c \arrow[r, "p"] \arrow[dr, "f"'] & S/E \arrow[d, "f'"] \\ & X \end{tikzcd}$$

Ceci signifie exactement que $(S/E, p)$ est un élément universel pour le foncteur $H : \mathcal{S}et \to \mathcal{S}et$ qui associe à tout ensemble X l'ensemble $H(X)$ de toutes les fonctions $f : S \to X$ pour lesquelles sEs' implique $f(s) = f(s')$.

Coproduits et colimites Pour toute catégorie $\mathcal{C}$ le foncteur diagonal $\Delta : \mathcal{C} \to \mathcal{C} \times \mathcal{C}$ est défini sur les objets par $\Delta(c) = (c, c)$ et sur les flèches par $\Delta(f) = (f, f)$.

Définition A.4.2 (coproduit) *Soit $\mathcal{C}$ une catégorie. Un* diagramme de coproduit *dans $\mathcal{C}$ est la donnée d'un objet $c \in |\mathcal{C}|$ et d'une flèche $(a, b) \to (c, c)$ de $\mathcal{C} \times \mathcal{C}$, i.e. une paire de flèches $(i : a \to c, j : b \to c)$ vérifiant la propriété suivante : pour toute paire $(f : a \to d, g : b \to d)$ il existe une unique flèche $f' : c \to d$ telle que $f = f' \circ i$ et $g = f' \circ j$.*

Il est clair que la flèche $(a, b) \to (c, c)$ est une flèche universelle de (a, b) vers Δ. Lorsqu'un tel diagramme coproduit existe, l'objet c est nécessairement unique (à isomorphisme près). On le note alors $c = a \coprod b$ ou $c = a + b$ et on l'appelle *objet coproduit*. Le diagramme coproduit est alors le suivant

$$a \xrightarrow{\ i\ } a \coprod b \xleftarrow{\ j\ } b$$

et les flèches i et j sont appelées les *injections* du coproduit $a \coprod b$. L'universalité de ce diagramme établit le fait que tout diagramme tel que le diagramme suivant peut être complété en h de façon unique de façon à être commutatif.

$$\begin{tikzcd} a \arrow[r, "i"] \arrow[dr, "f"'] & a \coprod b \arrow[d, "h"] & b \arrow[l, "j"'] \arrow[dl, "g"] \\ & d & \end{tikzcd}$$

Définition A.4.3 (pushout) *Soit $\mathcal{C}$ une catégorie. Étant donnée une paire $(f : a \to b, g : a \to c)$ de flèches de $\mathcal{C}$ de domaine commun $a \in |\mathcal{C}|$, un* pushout *de (f, g) est la donnée d'un diagramme commutatif tel que celui de gauche ci-dessous*

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[d, "g"'] & b \arrow[d, "u"] \\ c \arrow[r, "v"] & r \end{tikzcd} \qquad \begin{tikzcd} a \arrow[r, "f"] \arrow[d, "g"'] & b \arrow[d, "h"] \\ c \arrow[r, "k"] & s \end{tikzcd}$$

et tel que pour tout autre diagramme commutatif (comme celui de droite ci-dessus) construit sur (f, g), il existe une unique flèche $t \in \mathcal{C}(r, s)$ telle que $t \circ u = h$ et $t \circ v = k$.

Intuitivement, un pushout de (f, g) est la méthode universelle de compléter un diagramme commutatif construit sur (f, g).

Les coproduits et les pushouts sont en fait des cas particuliers d'un même concept : les colimites. Soit $\mathcal{C}, \mathcal{J}$ des catégories ($\mathcal{J}$ une catégorie d'index). Le foncteur diagonal $\Delta : \mathcal{C} \to \mathcal{C}^{\mathcal{J}}$ envoie chaque objet $c \in |\mathcal{C}|$ sur le foncteur $\Delta(c)$ qui associe la valeur c à tout objet $i \in |\mathcal{J}|$ et la valeur $\mathtt{Id}_{\mathtt{c}}$ à chaque flèche de J. Si $f : c \to c'$ est une flèche de $\mathcal{C}$, alors $\Delta(f)$ est la transformation naturelle $\Delta(f) : \Delta(c) \Rightarrow \Delta(c')$ qui a la même valeur f pour tout objet $i \in |\mathcal{J}|$. Tout foncteur $F : \mathcal{J} \to \mathcal{C}$ est un objet de $\mathcal{C}^{\mathcal{J}}$. Une flèche universelle (r, u) de F vers Δ est appelé une colimite de F.

Définition A.4.4 (colimite) *Un* diagramme de colimite *pour le foncteur F est la donnée d'un objet $r \in |\mathcal{C}|$, noté $\mathcal{C}o\mathcal{L}im F$, et d'une transformation naturelle $u : F \Rightarrow \Delta(r)$ universelle parmi les transformations naturelles $\tau : F \Rightarrow \Delta(c)$ $(c \in |\mathcal{C}|)$.*

Il est clair que (r, u) est une flèche universelle de F vers Δ. Puisque $\Delta(c)$ est le foncteur constant, les transformations naturelles τ consistent en la donnée d'une flèche $\tau_i \in \mathcal{C}(F_i, c)$ pour chaque objet $i \in |\mathcal{J}|$ telle que $\tau_j \circ F(u) = \tau_i$. Une telle transformation naturelle est habituellement appelée un *cône* de la base F vers le sommet c. La situation est telle que décrite par le diagramme suivant (pour un cas particulier de $\mathcal{J}$) où tous les triangles sont commutatifs.

$$\begin{tikzcd} F_i \arrow[r, "F_u"] \arrow[dr, "\tau_i"'] & F_j \arrow[r, "F_v"] \arrow[d, "\tau_j"] & F_k \arrow[dl, "\tau_k"] \\ & c & \end{tikzcd}$$

Une colimite de $F : \mathcal{J} \to \mathcal{C}$ consite alors en la donnée d'un objet $\mathcal{C}o\mathcal{L}imF \in |\mathcal{C}|$ et d'un cône $\mu : F \Rightarrow \Delta(\mathcal{C}o\mathcal{L}imF)$ universel, *i.e.* pour tout cône $\tau : F \Rightarrow \Delta(c)$ il existe une unique flèche $t' : \mathcal{C}o\mathcal{L}imF \to c$ telle que $\tau_i = t' \circ \mu_i$ pour tout index $i \in |\mathcal{J}|$. On dit alors que μ est le *cône limite* ou le *cône universel* (de la base F).

Produits et limites La notion de limite est une notion duale de celle de colimite définie ci-dessus. Soient donc $\mathcal{C}, \mathcal{J}$ deux catégories et soit $\Delta : \mathcal{C} \to \mathcal{C}^{\mathcal{J}}$ le foncteur diagonal.

Définition A.4.5 (limite) *Soit $F : \mathcal{J} \to, \mathcal{C}$ un foncteur. Une* limite *de F est la donnée d'un objet $r \in |\mathcal{C}|$, noté $\mathcal{L}imF$ et appelé* objet limite *du foncteur F, et d'une transformation naturelle $\nu : \Delta(r) \Rightarrow F$ universelle parmi les transformations naturelles $\tau : \Delta(c) \Rightarrow F$ $(c \in |\mathcal{C}|)$.*

Il est clair que (r, u) $(r \in |\mathcal{C}|$ et $u \in \mathcal{F})$ est une flèche universelle de Δ vers F. Puisque $\Delta(c) : \mathcal{J} \to \mathcal{C}$ est le foncteur constant, les transformations naturelles τ consistent en la donnée d'une flèche $\tau_i \in \mathcal{C}(c, \mathcal{F}_i)$ pour tout objet $i \in |\mathcal{J}|$ telle que pour toute flèche $u \in \mathcal{J}(i, j)$, on a $\tau_j = F(u) \circ \tau_i$. Une telle transformation naturelle est habituellement appelée un *cône* du sommet c vers la base F (ou cocône de la base F vers le sommet c). La situation est telle que décrite par le diagramme suivant (pour un cas particulier de $\mathcal{J}$) où tous les triangles sont commutatifs.

$$\begin{tikzcd}
F_i \arrow[r, "F_u"] & F_j \arrow[r, "F_v"] & F_k \\
& r \arrow[ul, "\tau_i"] \arrow[u, "\tau_j"] \arrow[ur, "\tau_k"'] &
\end{tikzcd}$$

Une limite de $F : \mathcal{J} \to \mathcal{C}$ consite alors en la donnée d'un objet $\mathcal{L}imF \in |\mathcal{C}|$ et d'un cône $\mu : \mathcal{L}imF \Rightarrow F$ universel, *i.e.* pour tout cône $\tau : c \Rightarrow F$ il existe une unique flèche $t : c \to \mathcal{L}imF$ telle que $\tau_i = \nu_i \circ t'$ pour tout index $i \in |\mathcal{J}|$. On dit alors que μ est le *cône limite* ou le *cône universel* (vers la base F).

Les propriétés de $\mathcal{L}im$ et $\mathcal{C}o\mathcal{L}im$ sont résumées dans le diagramme suivant,

$$\begin{tikzcd}
\mathcal{L}imF \arrow[r, "\nu"] & F \arrow[r, "\mu"] \arrow[d, equal] & \mathcal{C}o\mathcal{L}imF \arrow[d] \\
c \arrow[u] \arrow[r, "\tau"] & F \arrow[r, "\sigma"] & c
\end{tikzcd}$$

où les flèches horizontales sont des cônes et les flèches verticales sont des flèches de $\mathcal{C}$. Lorsque les limites existent il y a des isomorphismes naturels :

$$\begin{aligned}
\mathcal{C}(c, \mathcal{L}imF) \cong \quad & \mathcal{N}at(\Delta(c), F) & = \mathcal{C}one(c, F) \\
\mathcal{C}one(F, c) = \quad & \mathcal{N}at(F, \Delta(c)) & \cong \mathcal{C}(\mathcal{C}o\mathcal{L}imF, c)
\end{aligned}$$

Nous présentons maintenant quelques cas particuliers de la notion de limites. On notera que ce sont tous des notions duales des notions similaires définies pour les colimites.

Si $\mathcal{J}$ est la catégorie discrète $\{0, 1\}$, un foncteur $F : \{0, 1\} \to \mathcal{C}$ est la donnée d'une paire (a, b) de flèches de $\mathcal{C}$.

Définition A.4.6 (produit) *On appelle* produit *de a et b, noté $a \times b$ ou $a\Pi b$, l'objet limite pour le foncteur F ci-dessus. Le* diagramme limite *consiste en la donnée de l'objet limite $a \times b$ et de deux flèches p, q appelé* projections *du produit.*

$$a \xleftarrow{p} a \times b \xrightarrow{q} b$$

Définition A.4.7 (pullback) *Soit $\mathcal{C}$ une catégorie. Étant donnée une paire $(f : b \to a, g : c \to a)$ de flèche de $\mathcal{C}$ de codomaine commun $a \in |\mathcal{C}|$, un* pullback *de (f, g) est la donnée d'un diagramme commutatif tel que celui de gauche ci-dessous*

$$\begin{tikzcd} c \arrow[r, "k"] \arrow[d, "h"'] & d \arrow[d, "g"] \\ b \arrow[r, "f"] & a \end{tikzcd} \qquad \begin{tikzcd} e \arrow[r, "q"] \arrow[d, "p"'] & d \arrow[d, "g"] \\ b \arrow[r, "f"] & a \end{tikzcd}$$

et tel que pour tout autre diagramme commutatif (comme celui de droite ci-dessus) construit sur (f, g), il existe une unique flèche $t \in \mathcal{C}(e, c)$ telle que $q = k \circ t = h$ et $p = h \circ t$. L'objet c est souvent noté $a \times b$ ou $a\Pi b$ et est appelé pullback *ou* somme fibrée *de a et b.*

A.5 Adjoints

Proposition A.5.1 *Pour tout foncteur $U : \mathcal{A} \to \mathcal{X}$, les conditions suivantes sont équivalentes :*

1. *pour tout objet $X \in |\mathcal{X}|$ il existe une flèche universelle de X vers U ;*
2. *il existe un foncteur $F : \mathcal{X} \to \mathcal{A}$ et une bijection $\varphi_{X,A} : \mathcal{A}(F(X), A) \to \mathcal{X}(X, U(A))$ indexée par $|\mathcal{X}| \times |\mathcal{A}|$ naturelle en X et en A ;*
3. *il existe un foncteur $F : \mathcal{X} \to \mathcal{A}$, une transformation naturelle $\eta : \mathtt{Id}_{\mathcal{X}} \Rightarrow \mathtt{U} \circ \mathtt{F}$ (appelée* unité*) et une transformation naturelle $\epsilon : F \circ U \Rightarrow \mathtt{Id}_{\mathcal{A}}$ (appelée* co-unité*) telles que les équations triangulaires suivantes soient valides :*
   - $\epsilon(F) \circ F(\eta) = \mathtt{Id}_{\mathtt{F}}$,
   - $U(\epsilon) \circ \eta(U) = \mathtt{Id}_{\mathtt{U}}$.

Si les conditions ci-dessus sont vérifiées, U est appelé *adjoint à droite* et le foncteur F est appelé *adjoint à gauche à U*. Le quadruplet (U, F, η, ϵ) est appelé *adjonction* de la catégorie $\mathcal{X}$ à la catégorie $\mathcal{A}$.
Bien souvent, la notion d'adjonction est utilisée dans la forme « libre » suivante. Étant donnée une adjonction (U, F, η, ϵ), pour tout objet $X \in |\mathcal{X}|$, il existe un objet $F(X)$ que l'on dit *U-libre sur A* et une flèche $\eta(X) : X \to U(F(X))$ tels que pour tout objet $A \in |\mathcal{A}|$ et toute flèche $h : X \to U(A)$, il existe une unique flèche $h' : F(X) \to A$ telle que $h = \eta(x) \circ U(h')$.

$$\begin{tikzcd} X \arrow[rr, "\eta(X)"] \arrow[dr, "h"'] & & U(F(X)) \arrow[dl, "U(h')"] \\ & U(A) & \end{tikzcd} \qquad \begin{tikzcd} F(X) \arrow[d, "h'"] \\ A \end{tikzcd}$$

La proposition suivante donne l'une des propriétés les plus utiles des foncteurs adjoints.

Proposition A.5.2 *Les foncteurs adjoints à droite préservent les limites et, de façon duale, les foncteurs adjoints à gauche préservent les colimites.*

Les conditions équivalentes suivantes définissent un foncteur $U : \mathcal{X} \to \mathcal{X}'$ comme une équivalence de catégories.

Proposition A.5.3 *Pour tout foncteur $U : \mathcal{X} \to \mathcal{X}'$ les conditions suivantes sont équivalentes :*

1. *U appartient à une adjonction dont l'unité et la co-unité sont des isomorphismes naturels ;*
2. *U est* **full** *et* **faithfull** *et tout objet $A' \in |\mathcal{X}'|$ est isomorphe à $U(A)$, où $A \in |\mathcal{X}|$.*

Annexe B

Logique du premier ordre

Définition B.0.1 (signature) *Une* signature de premier ordre multi-sorte *est la donnée d'un triplet* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ *tel que :*

- $\mathcal{S}$ *est un ensemble de* noms de sortes *;*
- $\mathcal{F} = (\mathcal{F}_{\omega \to s})_{(\omega,s) \in \mathcal{S}^* \times \mathcal{S}}$ *est une famille d'ensembles de* noms de fonctions. *Pour tout couple* $(\omega, s) \in \mathcal{S}^* \times \mathcal{S}$, $\mathcal{F}_{\omega \to s}$ *est l'ensemble des noms de fonctions d'*arité ω *et de sorte* s*. Onnotera* $f : \omega \to s$ *tout le fait que le symbole* f *appartiennent à l'ensemble* $\mathcal{F}_{\omega,s}$*. L'ensemble* $\mathcal{F}_{\to s}$ *désigne l'ensemble des symboles de fonctions constantes de sorte* s*;*
- $\mathcal{R} = (\mathcal{R}_\omega)_{\omega \in \mathcal{S}^*}$ *est une famille d'ensembles de* noms de prédicats. *Pour tout* $\omega \in \mathcal{S}^*$, $\mathcal{R}_\omega$ *est l'ensemble des noms de prédicats d'*arité ω*. On notera* $r : \omega$ *le fait que le symbole* r *appartiennent à l'ensemble* $\mathcal{R}_\omega$*.*

Remarque B.0.1 *Dans la définition ci-dessus, lorsque la famille* $\mathcal{R}$ *est vide, on note* $(\mathcal{S}, \mathcal{F})$ *plutôt que* $(\mathcal{S}, \mathcal{F}, \mathcal{R})$ *et on parle de* signature algébrique.

Définition B.0.2 (morphisme) *Soient* $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ *et* $\Sigma' = (\mathcal{S}', \mathcal{F}', \mathcal{R}')$ *deux signatures. Un* morphisme de signatures $\sigma : \Sigma \to \Sigma'$ *est la donnée d'un triplet* $(\sigma_\mathcal{S}, \sigma_\mathcal{F}, \sigma_\mathcal{R})$ *d'applications telles que* $\sigma_\mathcal{S} : \mathcal{S} \to \mathcal{S}'$, $\sigma_\mathcal{F} : \mathcal{F} \to \mathcal{F}'$, $\sigma_\mathcal{R} : \mathcal{R} \to \mathcal{R}'$ *et telles que la condition suivante soit vérifiée :*

$$\forall (\omega, s) \in \mathcal{S}^* \times \mathcal{S}, \forall f \in \mathcal{F}_{\omega \to s},\ \sigma_\mathcal{F}(f) \in \mathcal{F}'_{\sigma_\mathcal{S}(\omega) \to \sigma_\mathcal{S}(s)}$$

où $\sigma_\mathcal{S}(\omega) = \sigma_\mathcal{S}(s_1) \ldots \sigma_\mathcal{S}(s_n)$ *pour tout* $\omega = s_1 \ldots s_n \in \mathcal{S}^*$*.*

Pour construire l'ensemble des termes, on considère maintenant la donnée d'une famille $\mathcal{X} = (\mathcal{X}_s)_{s \in \mathcal{S}}$ d'ensembles de variables indexés par les sortes. On suppose de plus que pour toute signature $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$, on a $\mathcal{X} \cap (\mathcal{F} \cup \mathcal{R}) = \varnothing$.

Définition B.0.3 (Σ-terme) *Soient* $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ *une signature et* $\mathcal{X} = (\mathcal{X}_s)_{s \in \mathcal{S}}$ *une famille d'ensembles de variables. Soit* $s \in \mathcal{S}$*. L'ensemble des* Σ-termes avec variables de sorte s, *noté* $T_\Sigma(\mathcal{X})_s$, *est le plus petit ensemble au sens de l'inclusion satisfaisant les propriétés suivantes :*

- $\mathcal{X}_s \subseteq T_\Sigma(\mathcal{X})_s$ *;*
- $\forall \omega \in \mathcal{S}^*, \forall f \in \mathcal{F}_{\omega \to s}, \forall (t_1, \ldots, t_n) \in T_\Sigma(\mathcal{X})_\omega$,

$$f(t_1, \ldots, t_n) \in T_\Sigma(\mathcal{X})_s$$

où $T_\Sigma(\mathcal{X})_\omega = T_\Sigma(\mathcal{X})_{s_1} \times \cdots \times T_\Sigma(\mathcal{X})_{s_n}$ *pour tout* $\omega = s_1 \ldots s_n$*.*

On note $T_\Sigma(\mathcal{X})$ l'ensemble de tous les Σ-termes construit au dessus de Σ.

On étend naturellement les morphismes de signatures aux ensembles de termes.

Définition B.0.4 (extension aux termes) *Soit $\sigma : \Sigma \to \Sigma'$ un morphisme de signatures. On étend σ en une application $\sigma^\sharp : T_\Sigma(\mathcal{X}) \to T_{\Sigma'}(\mathcal{X}')$ inductivement de la façon suivante :*

- *si $t \in T_\Sigma(\mathcal{X})_s$ est une variable ou un symbole de fonction d'arité ϵ, alors $\sigma^\sharp(t) = \sigma(t)$;*
- *si $t \in T_{\Sigma_1}(\mathcal{X}_1)$ est un Σ_1-terme de la forme $f(t_1, \ldots, t_n)$, alors $\sigma^\sharp(t) = \sigma(f)(\sigma^\sharp(t_1), \ldots, \sigma^\sharp(t_n))$.*

Lemme B.0.1 *Soit $\sigma : \Sigma \to \Sigma'$ un morphisme de signatures. L'application $\sigma^\sharp : T_\Sigma(\mathcal{X}) \to T_{\Sigma'}(\mathcal{X}')$ est l'unique extension du morphisme σ aux Σ_1-termes.*

Preuve Il est clair, par construction, que $\sigma^\sharp$ existe. Son unicité est tout autant immédiate. □

Définition B.0.5 (Σ-formule atomique) *Soit $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ une signature. L'ensemble des* Σ-formules atomiques, *noté $\mathcal{A}tom_\Sigma$ est le plus petit ensemble au sens de l'inclusion satisfaisant la propriété suivante :*

$$\forall \omega \in \mathcal{S}^*, \forall r \in \mathcal{R}_\omega, \forall (t_1, \ldots, t_n) \in T_\Sigma(\mathcal{X})_\omega,\ r(t_1, \ldots, t_n) \in \mathcal{A}tom_\Sigma$$

Définition B.0.6 (Σ-structure du 1^{er}ordre) *Soit $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ une signature. Une* Σ-structure du premier ordre *$\mathcal{M}$ est la donnée :*

- *pour tout symbole de sorte $s \in \mathcal{S}$ d'un ensemble M_s ;*
- *pour tout symbole de fonction $f \in \mathcal{F}_{\omega \to s}$ d'une fonction $f^\mathcal{S} : M_\omega \to M_s$, où $M_\omega = M_{s_1} \times \cdots \times M_{s_n}$ pour tout $\omega = s_1 \ldots s_n$;*
- *pour tout symbole de prédicat $r \in \mathcal{F}_\omega$ d'une relation $r^\mathcal{S} \subseteq M_\omega$.*

Remarque B.0.2 *Lorsqu'on considère des signatures algébriques, le dernier point de la définition ci-dessus n'est d'aucun usage puisque les signatures ne contiennent pas de symboles de prédicats. Les structures ainsi formées sont donc la donnée d'un ensemble de base M_s pour chaque symbole de sorte $s \in \mathcal{S}$ et d'une fonction $f^\mathcal{S} : M_\omega \to M_s$ pour chaque symbole de fonction d'arité ω et de sorte s. On appelle Σ-algèbre une telle structure.*

Pour interpréter les termes et les formules dans une structure donnée, il est nécessaire de donner en premier lieu une valeur aux variables dans cette structure.

Définition B.0.7 (interprétation de variables) *Soient $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ une signature, $\mathcal{X} = (\mathcal{X}_s)_{s \in \mathcal{S}}$ et $\mathcal{M}$ une Σ-structure du premier ordre. Une* interprétation des variables *$\nu : \mathcal{X} \to \mathcal{M}$ est la donnée, pour chaque symbole de sorte $s \in \mathcal{S}$, d'une application $\nu_s : \mathcal{X}_s \to M_s$.*

Définition B.0.8 (extension aux termes) *Soient $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ une signature, $\mathcal{X} = (\mathcal{X}_s)_{s \in \mathcal{S}}$ et $\mathcal{M}$ une Σ-structure du premier ordre. Soit $\nu : \mathcal{X} \to M$ une interprétation de variables. On étend ν en une application $\nu^\sharp : T_\Sigma(\mathcal{X}) \to M$ inductivement de la façon suivante :*

- *si $t \in T_{\Sigma_1}(\mathcal{X}_1)$ est une variable, alors $\nu^\sharp(t) = \nu(t)$;*

- *si* $t \in T_{\Sigma_1}(\mathcal{X}_1)$ *est un* Σ_1*-terme de la forme* $f(t_1, \ldots, t_n)$*, alors* $\nu^\sharp(t) = f^{\mathcal{M}}(\nu^\sharp(t_1), \ldots, \nu^\sharp(t_n))$.

Lemme B.0.2 *Soit* $\nu : \mathcal{X} \to M$ *une interprétation de variables. L'application* $\nu^\sharp : T_{\Sigma_1}(\mathcal{X}_1) \to M$ *est l'unique extension de l'application* ν *aux* Σ_1*-termes.*

Preuve Il est clair, par construction, que $\nu^\sharp$ existe. Son unicité est tout autant immédiate. □

Définition B.0.9 (homomorphisme) *Soient* $\Sigma = (\mathcal{S}, \mathcal{F}, \mathcal{R})$ *et une signature du premier ordre et* $\mathcal{M}, \mathcal{N}$ *des* Σ*-structures du premier ordre. Un* homomorphisme de Σ-structures $\mu : \mathcal{M}_1 \to \mathcal{M}_2$ *est la donnée :*

- *d'une application* $\mu_s : M_s \to N_s$ *pour chaque symbole de sorte* $s \in \mathcal{S}$ *;*
- *d'une application* $\mu_{\mathcal{F}} : \mathcal{F}^{\mathcal{M}} \to \mathcal{F}^{\mathcal{N}}$*, où* $\mathcal{F}^{\mathcal{M}} = \{f^{\mathcal{M}} / f \in \mathcal{F}\}$ *(resp.* $\mathcal{F}^{\mathcal{N}} = \{f^{\mathcal{N}} / f \in \mathcal{F}\}$*), telle que pour tout symbole de fonction* $f \in \mathcal{F}_{\omega,s}$*,* $\mu_{\mathcal{F}}(f^{\mathcal{M}}) = f^{\mathcal{N}}$
- *d'une application* $\mu_{\mathcal{R}} : \mathcal{R}^{\mathcal{M}} \to \mathcal{R}^{\mathcal{N}}$*, où* $\mathcal{R}^{\mathcal{M}} = \{r^{\mathcal{M}} / r \in \mathcal{F}\}$ *(resp.* $\mathcal{R}^{\mathcal{N}} = \{r^{\mathcal{N}} / r \in \mathcal{R}\}$*), telle que pour tout symbole de prédicat* $r \in \mathcal{R}_{\omega,s}$*,* $\mu_{\mathcal{R}}(r^{\mathcal{M}}) = r^{\mathcal{N}}$.

Annexe C

Institutions stratifiées avec constructeurs de formules

On introduit les deux opérations suivantes sur la classe des signatures de constructeurs munie des morphismes de constructeurs :

identité : à toute signature de constructeurs $\mathcal{C} = (\mathcal{C}^n)_{n \in \mathbb{N}}$, on associe une application, notée $\mathtt{Id}_{\mathcal{C}}$, définie par :

- $\mathtt{dom}(\mathtt{Id}_{\mathcal{C}}) = \mathtt{codom}(\mathtt{Id}_{\mathcal{C}}) = \mathcal{C}$,
- $\forall n \in \mathbb{N}, \forall c \in \mathcal{C}^n$, $\mathtt{Id}_{\mathcal{C}}(\mathtt{c}) = \mathtt{c}$.

Il est clair que $\mathtt{Id}_{\mathcal{C}}$ défini bien un morphisme de constructeurs que l'on appellera *identité sur $\mathcal{C}$*.

composition : à tout couple $(\varsigma_1, \varsigma_2)$ de morphismes de signatures de constructeurs tels que $\mathtt{dom}(\varsigma_2) = \mathtt{codom}(\varsigma_1)$ on associe une application, notée $\varsigma_2 \circ \varsigma_1$, définie par :

- $\mathtt{dom}(\varsigma_2 \circ \varsigma_1) = \mathtt{dom}(\varsigma_1)$,
- $\mathtt{codom}(\varsigma_2 \circ \varsigma_1) = \mathtt{codom}(\varsigma_2)$,
- $\forall c \in \mathtt{dom}(\varsigma_1)$, $\varsigma_2 \circ \varsigma_1(c) = \varsigma_2(\varsigma_1(c))$.

La composition de deux morphismes de constructeurs défini bien un morphisme de constructeurs.

Preuve À faire $\square$ On appelle *composée de ς_1 et ς_2* le morphisme de constructeurs ainsi défini.

Proposition C.0.4 (catégorie de constructeurs) *La classe des signatures de constructeurs introduites à la définition 3.1.1, partie II munie des morphismes introduis à la définition 3.1.2, partie II est une catégorie.*

Preuve Découle directement de la définition des opérations identité et composition définies ci-dessus. $\square$

Soit $\mathcal{C} \in |\mathit{LogSig}|$ une signature de constructeurs. On introduit les deux opérations suivantes sur la classe des $\mathcal{C}$-algèbre munie des morphismes de $\mathcal{C}$-algèbres :

identité : à toute $\mathcal{C}$-algèbre $\mathcal{A}$, on associe une application, notée $\mathtt{Id}_{\mathcal{A}}$, définie par :

- $\mathtt{dom}(\mathtt{Id}_{\mathcal{A}}) = \mathtt{codom}(\mathtt{Id}_{\mathcal{A}}) = \mathcal{A}$,

- $\forall n \in \mathbb{N}, \forall c^n \in \mathcal{C}$, $\mathtt{Id}_{\mathcal{A}}(\mathtt{c}^{\mathcal{A}}) = \mathtt{c}^{\mathcal{A}}$.

Il est clair que $\mathtt{Id}_{\mathcal{A}}$ défini bien un morphisme de $\mathcal{C}$-algèbres que l'on appellera *identité sur* $\mathcal{A}$.

composition: à tout couple (μ_1, μ_2) de morphismes de $\mathcal{C}$-algèbres tels que $\mathtt{dom}(\varsigma_2) = \mathtt{codom}(\varsigma_1)$ on associe une application, notée $\mu_2 \circ \mu_1$, définie par :

- $\mathtt{dom}(\mu_2 \circ \mu_1) = \mathtt{dom}(\mu_1)$,
- $\mathtt{codom}(\mu_2 \circ \mu_1) = \mathtt{codom}(\mu_2)$,
- $\forall c \in \mathtt{dom}(\mu_1)$, $\mu_2 \circ \mu_1(c) = \mu_2(\mu_1(c))$.

La composition de deux morphismes de $\mathcal{C}$-algèbres défini bien un morphisme de $\mathcal{C}$-algèbres.

Preuve À faire □ On appelle *composée de μ_1 et μ_2* le morphisme de $\mathcal{C}$-algèbres ainsi défini.

Proposition C.0.5 (catégorie de $\mathcal{C}$-algèbres) *La classe des mathcalC-algèbres introduites à la définition 3.1.3, partie II munie des morphismes introduis à la définition 3.1.4, partie II est une catégorie.*

Preuve Découle directement de la définition des opérations identité et composition définies ci-dessus. □

Index

Bibliographie

[1] J.-R. Abrial. *The B Book - Assigning Programms to Meanings.* Cambridge University Press, 1996.

[2] M. Aiguier. *Spécifications algébriques par objets : une proposition de formalisme et ses applications à l'implantation abstraite.* PhD thesis, Université Paris XI, Orsay, 1995.

[3] M. Aiguier. Étoile-Spécifications, an Object-Oriented Formalism With Refinement. Technical report, UNiversité d'Évry, 2002.

[4] M. Aiguier and G. Bernot. *Information Systems Correctness and Reusability*, chapter Algebraic Semantics of Object Type Specifications. World Scientific Publishing, 1995.

[5] M. Aiguier, C. Gaston, and P. Le Gall. Feature Specification: a Logic-Independent Approach. Technical report, Université d'Évry, 2004.

[6] H. Andréka and I. Németi. Loś Lemma Holds in Every Category. *Studi Scientiarum Mathematicarum Hungarica*, 1978.

[7] A. Arnould. *Test à partir de spécifications de structures bornées : une théorie du test, une méthode de sélection, un outil d'assistance à la selection.* PhD thesis, Université de Paris-Sud, 1997.

[8] M. Arrais and J.L. Fiadeiro. Unifying Theories in Different Institutions. In M. Haveraaen, O. Owe, and O.-J. Dahl, editors, *Recent Trends in Data Type Specification*, number 1130 in Lecture Notes in Computer Science, pages 81–101. Springer Verlag, 1996.

[9] D. Bahrami. *Une axiomatisation de la réécriture abstraite.* PhD thesis, LaMI-Université d'Évry, 2003.

[10] M. Barr and C. Wells. *Category Theory for Computing Science.* Prentice-Hall, 1990.

[11] J Barwise. Axioms for Abstract Model Theory. *Annals of Mathematical Logic*, 7:221–265, 1974.

[12] J. Barwise and S. Feferman, editors. *Model-Theoretic Logics.* Springer Verlag, 1985.

[13] J. Bergstra, J. Heering, and P. Klint. Module Algebra. *Journal of the Association for Computing Machinery (ACM)*, 37(2):335–372, 1990.

[14] J.A. Bergstra, A. Ponse, and S.A. Smolka, editors. *Handbook of Process Algebra.* North Holland, 2001.

[15] W.E. Beth. On Padoa's Method in the Theory of Definitions. *Indag. Math.*, 1953.

[16] M. Bidoit. The Stratified Loose Approach: a Generalisation of Initial and Loose Semantics. In *Recent Trends in Algebraic Development Techniques*, Lecture Notes in Computer Science. Springer Verlag, 1987.

[17] M. Bidoit and A. Tarlecki. Behavioural Satisfacton and Equivalence in Concrete Model Categories. In H. Kirchner, editor, *Trees in Algebra and Programming - CAAP'96*, volume 1059 of *Lecture Notes in Computer Science*, pages 241–256. Springer Verlag, 1996.

[18] T. Borzyszkowski. Completness of the Logical System for Structured Specifications. In F. Parisi-Presice, editor, *Workshop on Algebraic Development Techniques (WADT'97)*, Recent Trends in Algebraic Development Techniques, Selected Papers, pages 107–121, 1997.

[19] T. Borzyszkowski. Moving Specification Structures Between Logical Systems. In *Proceedings of the Workshop on Algebraic Development Techniques (WADT'98)*, pages 16–30, 1998.

[20] T. Borzyszkowski. Generalized Interpolation in CASL. *Information Processing letters*, 2001.

[21] T. Borzyszkowski. Logical Systems for Structured Specifications. *Theoretical Computer Science*, 286:197–245, 2002.

[22] R. Burstall and J. Goguen. The Semantics of CLEAR, a Specification Language. In D. Björner, editor, *Proceedings of the Winter School on Abstract Software Specification (WSASS'80)*, volume 86 of *Lecture Notes in Computer Science*, pages 292–332. Springer Verlag, 1980.

[23] C. Caleiro. *Combining Logics.* PhD thesis, Universidade Técnica de Lisboa, Instituto Superior Técnico, 2000.

[24] C. Caleiro, W. Carnielli, M. Coniglio, A. Sernadas, and C. Sernadas. Fibring Non-Truth-Fonctional Logics: Completness Preservation. *Journal of Logic, Language and Information*, 12(2):183–211, 2003.

[25] C. Caleiro, P. Gouveia, and J. Ramos. Completeness Results for Fibred Parchments: Beyond the Propositional Base. In *Recent Trends in Algebraic Development Techniques*, volume 2755 of *Lecture Notes in Computer Science*, pages 185–200. Springer Verlag, 2003.

[26] C. Caleiro, P. Mateus, J. Ramos, and A. Sernadas. Combining Logics: Parchments Revisited. In *Recent Trends in Algebraic Development Techniques*, volume 2267 of *Lecture Notes in Computer Science*, pages 48–70. Springer Verlag, 2001.

[27] M. V. Cengarle. *Formal Specifications With High-Order Parametrization.* PhD thesis, Institute für Informatik, Ludwig-Maximilians-Universität, Müenchen, 1994.

[28] M. Cerioli. *Relationships Between Logical Formalisms.* PhD thesis, Università di Pisa-Genova-Udine, 1993.

[29] M. Cerioli and J. Meseguer. May I Borrow Your Logic? (Transporting Logical Structures along Maps). *Theoretical Computer Science*, 173(2):311–347, 1997.

[30] C.C. Chang and H.J. Keisler. *Model Theory*, volume 73 of *Studies in Logic and Foundations of Mathematics.* North Holland, 1973.

[31] M. E. Coniglio, A. Sernadas, and C. Sernadas. Fibring Logics with Topos Semantics, 2002.

[32] M.E. Coniglio, A.T. Martins, A. Sernadas, and C. Sernadas. Fibring (Para)consistent Logics. Technical report, CMA/IST, 2000.

[33] R. Cori and D. Lascar. *Logique mathématique*, volume 2 of *Axiomes*. Masson, 1993.

[34] W. Craig. Linear Reasoning: A New Form of Herbrand-Gentzen Theorem. *Journal of Symbolic Logic*, 22(250-268), 1957.

[35] W. Craig. Three Uses of Herbrand-Gentzen Theorem in Relating Model Theory and Proof Theory. *Journal of Symbolic Logic*, 22:269–285, 1957.

[36] R. Diaconescu. Extra Theory Morphisms For Institutions: Logical Semantics For Multi-Paradigm Languages. *Applied Categorical Structures*, 1998.

[37] R. Diaconescu. Grothendieck Institutions. *Applied Categorical Structures*, 10(4):383–402, 2002.

[38] R. Diaconescu. Herbrand Theorems in Arbitrary Institutions. *Information Processing Letters*, 90(1):29–39, 2002.

[39] R. Diaconescu. Institution-Independent Iltraproducts. *Fundamenta Informaticae*, 55(3-4):321–348, 2003.

[40] R. Diaconescu. Elementary Diagrams in Institutions. *Journal of Logic and Computation*, 14(02-37), 2004.

[41] R. Diaconescu. An Institution-Independent Proof of Craig Interpolation Property. *Studia Logica*, 76(3), 2004.

[42] R. Diaconescu. Interpolation in Grothendieck Institutions. *Theoretical Computer Science*, 55(3-4):321–348, 2004.

[43] R. Diaconescu. *Institution-Independant Model Theory.* 2005. Ouvrage en cours de rédaction.

[44] R. Diaconescu and K. Futatsugi. Logical Foundations of CafeOBJ. *Theoretical Computer Science*, 285:289–318, 2002.

[45] R. Diaconescu, J.A. Goguen, and P. Stefaneas. Logical Support for Modularisation. In G. Huet and G. Plotkin, editors, *Logical Environment*, pages 83–130, 1993.

[46] T. Dimitrakos and T. Maibaum. Notes on Refinement, Interpolation and Uniformity. In *Proceedings of the conference on Automated Software Engineering (ASE'97)*, 1997.

[47] T. Dimitrakos and T. Maibaum. On a Generalized Modularization Theorem. *Information Processing Letter*, 2000.

[48] H.-D. Ehrich. On the Theory of Specification, Implementation and Parameterization of Abstract Data Types. *Journal of the ACM*, 29(1), 1982.

[49] H. Ehrig and H. Kreowski. *Algebraic Foundations of Systems Specification*, chapter Refinement and Implementation. IFIP State-of-the-Art Reports. Springer Verlag, 1999.

[50] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specifications I: Equations and Initial Semantics*. Springer Verlag, 1985.

[51] J. Farrés Casals. *Verification in ASL and Related Specification Languages*. PhD thesis, Department of Computer Science, University of Edinburgh, 1992.

[52] J.L. Fiadeiro and A. Sernadas. Structuring Theories on Consequence. In D. Sannella and A. Tarlecki, editors, *Recent Trends in Data Type Specification*, pages 44–72. Springer Verlag, 1988.

[53] M. Finger and D. Gabbay. Adding a Temporal Dimension to a Logic System. *Journal of Logic, Language, and Information*, 1(3):203–233, 1992.

[54] M. Finger and D. Gabbay. Combining Temporal Logic Systems. *Notre Dame Journal of Logic*, 37(2):204–232, 1996.

[55] J. Flum. *Model-Theoretic Logic*, chapter Characterizing Logics, pages 77–121. Perspectives in Mathematical Logic. Springer Verlag, 1985.

[56] D. Gabbay. Fibred Semantics and the Weaving of Logics. Part 1. Modal and Intuitionistic Logics. *Journal of Symbolic Logic*, 61(4):1057–1120, 1996.

[57] D. Gabbay. An Overview of Fibred Semantics and the Combination of Logics. In F. Baader and K. Schultz, editors, *Proceedings of Frontiers of Combining Systems (FroCoS'96)*, pages 1–55. Kluwer Academic Publishers, 1996.

[58] D. Gabbay. *Fibring Logics.* Clarendon Press - Oxford, 1999.

[59] D. Gabbay and V. Shehtman. Products of Modal Logics, part 1. *Logic Journal of the IGPL*, 6:73–146, 1998.

[60] D.M. Gabbay. Semantic Proof of the Craig Interpolation Theorem for Intuitionistic Logic and Extensions. In R.O. Gandy and C.M.Y. Yates, editors, *Proceedings of the Logic Colloquium'69*, 1971.

[61] J. Goguen and R. Burstall. Introducing Institutions. In E. Clarke and D. Kozen, editors, *Proceedings of the Logics of Programming Workshop*, volume 164 of *LNCS*, pages 221–256. Springer Verlag, 1984.

[62] J. Goguen and R. Burstall. A Study in the Foundation of Programming Methodology: Specifications, Institutions, Charters and Parchments. In D Pitt, S. Abramsky, A. Poigné, and D. Rydeheard, editors, *Proceedings of the Conference on Category Theory and Computer Programming*, volume 240 of *Lecture Notes in Computer Science*, pages 313–333. Springer Verlag, 1985.

[63] J. Goguen and R. Burstall. Institutions: Abstract Model Theory for Specification and Programming. *Journal of the Association for Computing Machinery*, 39(1):95–146, 1992.

[64] R. Goldblatt. *Topoi, the Categorical Analysis of Logic.* North Holland, 1979.

[65] G. Grätzer. *Universal Algebra.* Springer Verlag, 1979.

[66] D. Găină and A. Popescu. Elementary Morphisms in Institutions. submitted for publication.

[67] R. Harper, D. Sannella, and A. Tarlecki. Structured Theory Presentations and Logic Representations. *Annals of Pure and Applied Logic*, 67(1-3):113–160, 1994.

[68] R. Hennicker and M. Bidoit. Observational logic. In *Proceedings of the Conference on Algebraic Methodology and Software Technology 1998 (AMAST'98)*, volume 1548 of *Lecture Notes in Computer Science*, 1999.

[69] M.-R. Huth and M.D. Ryan. *Logic in Computer Science: Modelling and Reasoning about Systems.* Cambridge University Press, 2000.

[70] C.B. Jones. *Systematic Software Development Using VDM.* Prentice Hall, 1990.

[71] A. Kock and G.-E. Reyes. *Handbook of Mathematical Logic*, chapter Doctrines in Categorcal Logic, pages 283–313. North Holland, 1977.

[72] M. Kracht and F. Wolter. Properties of Independently Axiomatizable Bi-Modal Logics. *Journal of Symbolic Logic*, 56(4):1496–1485, 1991.

[73] H.-J. Kreowski and T. Mossakowski. Equivalence and Difference Between Institutions: Simulating Horn Clause Logic with Based Algebras. *Mathematical Structures in Computer Science*, 5(2):189–215, 1995.

[74] Á. Kurucz. On Axiomatising Products of Kripke Frames. *Journal of Symbolic Logic*, 65:923–945, 2000.

[75] F.-W. Lawvere. Toposes, Algebraic Geometry and Logic. *Lecture Notes in Computer Science*, 274, 1972.

[76] F.-W. Lawvere. Introduction to Part i. In F.-W. Lawvere, C. Maurer, and G.-C. Wraith, editors, *Model Theory and Topoi*, volume 445, pages 3–14. Springer Verlag, 1975.

[77] P. Le Gall and A. Arnould. Formal Specifications and Test: Correctness and Oracle. In *COMPASS/ADT*, pages 342–358, 1995.

[78] S. Mac Lane. *Categories for the Working Mathematician.* Graduate Texts in Mathematics. Springer Verlag, 1971.

[79] B. Mahr and J.A. Makowsky. An Axiomatic Approach to Semantics of Specification Languages. In *Proceedings of GI-conference on Theoretical Computer Science*, number 145 in Lecture Notes in Computer Science. Springer Verlag, 1984.

[80] B. Mahr and J.A. Makowsky. Characterising Specification Languages Wich Admit Initial Semantics. In *Proceedings of CAAP'83*, number 159 in Lecture Notes in Computer Science, pages 300–316. Springer Verlag, 1984.

[81] B. Mahr and J.A. Makowsky. Characterising Specification Languages Wich Admit Initial Semantics. *Theoretical Computer Science*, 31:49–59, 1984.

[82] T. Maibaum and M.R. Sadler. Axiomatising Specification Theory. In H.-J. Kreowski, editor, *Recent Trends in Data Type Specification*, pages 171–177. Springer Verlag, 1985.

[83] T. Maibaum, P. Veloso, and M. Sadler. *Foundations of Software Technology and Theoretical Computer Science*, chapter Logical Specification and Implementation, pages 13–30. Springer Verlag, 1984.

[84] M. Makkai. On Gabbay's Proof of the Craig Interpolation Theorem for Intuitionistic Logic. *Notre Dame Journal of Logic*, 36(3), 1995.

[85] L. Maksimova. Craig's Interpolation Theorem and Amalgamable Varieties. *Soviet. Math. Dokl*, 18(6):1550–1553, 1977.

[86] A. Martini and U. Wolter. A Systematic Study of Mappings Between Institutions. In *Workshop on Algebraic Development Techniques*, pages 300–315, 1997.

[87] M. MArx and C. Areces. Failure of Interpolation in Combined Modal Logics. *Notre Dame Journal of Formal Logic*, 39(2):253–272, 1998.

[88] M. Marx and M. Reynolds. Undecidability of Compass Logic. *Journal of Logic and Computation*, 9(6):897–914, 1999.

[89] G. Matthiessen. Regular and Strongly Finitary Structures Over Strongly Algebroidal Categories. *Canadian Journal of Mathematics*, 1978.

[90] J. Meseguer. General Logics. In Ebbinghaus H.-D. et al., editor, *Proceedings of the Logic Colloquium*, pages 275–329. North Holland, 1989.

[91] J. Meseguer. Membership Algebra as a Logical Framework for Equational Specification. In F. Parisi-Pressice, editor, *Proceedings of the Workshop on Algebraic Description Techniques (WADT'97)*, pages 18–61, 1998.

[92] G. Mints. Modularisation and Interpolation, 1998.

[93] J. Mossakowski, T.and Goguen, R. Diaconescu, and A. Tarlecki. What is a Logic? In J.-Y. Beziau, editor, *Proceedings of UNILOG'05*, 2005.

[94] T. Mossakowski. Using Limits of Parchments to Systematically Construct Institutions of Partial Algebras. In M. Haveraaen, O. Owe, and O.-J. Dahl, editors, *Recent Trends in Data Type Specification*, volume 1130 of *Lecture Notes in Computer Science*, pages 379–393. Springer Verlag, 1996.

[95] T. Mossakowski. Relating CASL With Other Specification Languages: the Institution Level. *Theoretical Computer Science*, 286:367–475, 2002.

[96] T. Mossakowski. *Heterogeneous Specification and the Heterogeneous Tool Set*. PhD thesis, Bremen University, 2005.

[97] T. Mossakowski, A. Tarlecki, and W. Pawlowski. Combining and Representing Logical Systems. In *Category Theory and Computer Science*, volume 1290, pages 177–196. Springer Verlag, 1997.

[98] T. Mossakowski, A. Tarlecki, and W. Pawlowski. Combining and Representing Logical Systems Using Model-Theoretic Parchments. In *Recent Trends in Abstract Data Type*, volume 1376 of *Lecture Notes in Computer Science*, pages 349–364. Springer Verlag, 1998.

[99] P.D. Mosses. CoFI: the Common Framework Initiative for Algebric Specification and Development. In *Theory and Practice of Software Development*, volume 1214 of *LNCS*, pages 115–137. Springer Verlag, 1997.

[100] D. Mundici. Complexity of Craig Interpolation. *Fundamentae Informaticae*, 5(3-4):261–278, 1982.

[101] D. Mundici. NP and Craig's Interpolation Theorem. In *Preceedings of Logic Colloquium'82*, volume 112 of *Stud. Log. Found. Math.*, pages 345–358. North Holland, 2002.

[102] Object Modelling Group. *Unified Modelling Langage Specification, version 1.5*, mars 2003.

[103] The CoFI Task Group on Semantics. CASL the Common Algebraic Specification Language - Semantics, 1999. Version 1.0 [1].

[104] F. Orejas. *Algebraic Foundations of Systems Specification*, chapter Structuring and Modularity. IFIP State-of-the-Art Reports. Springer Verlag, 1999.

[105] D.L. Parnas. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15(12):10531058, december 1972.

[106] W. Pawlowski. Context Parchments. In M. Haveraaen, O. Owe, and O.-J. Dahl, editors, *Recent Trends in Abstract Data Type*, volume 1376 of *Lecture Notes in Computer Science*. Springer Verlag, 1998.

[107] A.-M Pitts. *Handbook of Logic in Computer Science - Logic and Algebraic Methods*, volume 5, chapter Categorical Logic. Oxford University Press, 2000.

[108] W. Reisig. *Petri Nets: an Introduction.* EATCS Monographs on Theoretical Computer Science. Springer Verlag, 1985.

1. http://www.brics.dk/Projects/CoFI/Notes/S-9 et ftp://ftp.brics.dk/Projects/CoFI/Notes/S-9

[109] A. Robinson. A Result in Consistency and Its Applications to the Theory of Definition. *Indag. Math.*, 18:47–58, 1956.

[110] P.-H. Rodenburg. A Simple Algebraic Proof of the Equational Interpolation Theorem. *Algebra Universalis*, 28:48–51, 1991.

[111] P. H. Rodenburg and R. J. van Glabbeek. An Interpolation Theorem in Equationnal Logic. Technical report, Department of Computer Science, Center for Mathematics and Computer Science, Amsterdam, 1988.

[112] A. Salibra and G. Scollo. A Soft Stairway to Institutions. In *Recent Trends in Data Type Specification*, volume 655 of *Lecture Notes in Computer Science*, pages 310–329. Springer Verlag, 1992.

[113] A. Salibra and G. Scollo. Compactness and Löwenheim-Skolem Properties in Pre-Institution Categories. In C.Rauszer, editor, *Algebraic Methods in Logic and in Computer Science*, volume 28, pages 67–94. Banach Center Publications, 1993.

[114] A. Salibra and G. Scollo. Interpolation and Compactness in Categories of Pre-Institutions. *Mathematical Structures in Computer Science*, 6(3):261–286, June 1996.

[115] D. Sannella, S. Sokolowski, and A. Tarlecki. Towards Formal Developement of Programs From Algebraic Specifications: Parametrization Revisited. *Acta Informatica*, 29:689–736, 1992.

[116] D. Sannella and A Tarlecki. Specifications in an Arbitrary Institution. *Information and Computation*, 76:165–210, 1988.

[117] D. Sannella and A Tarlecki. Toward Formal Development of Programs From Algebraic Specifications: Implementations Revisited. *Acta Informatica*, 25:233–281, 1988.

[118] D. Sannella and A Tarlecki. Toward Formal Development of Programs From Algebraic Specifications: Model-Theoretic Foundations. In W. Kuich, editor, *Automata, Languages and Programming: Proc. of the 19th International Colloquium*, pages 656–671. Springer Verlag, 1992.

[119] D. Sannella and A Tarlecki. Algebraic Specification and Formal Methods for Programming Development: What Are the Real Problems? In *Current Trends in Theoretical Computer Science. Essays and Tutorials*, volume 40 of *World Scientific Series in Computer Science*, pages 115–120. Rozenberg, G. and Salomaa, A., 1993.

[120] D. Sannella and A Tarlecki. Model-Theoretic Foundations for Formal Program Development: Basic Concepts and Motivation. Technical report, Institute of Computer Science PAS, Warsaw, 1995.

[121] D. Sannella and A. Tarlecki. Essential Concepts of Algebraic Specification and Program Development. *Formal Aspects of Computing*, 9(3):229–269, 1997.

[122] A. Sernadas, C. Sernadas, and C. Caleiro. Fibring of Logics as a Categorial Construction. *Journal of Logic and Computation*, 9(2):149–179, 1999.

[123] A. Sernadas, C. Sernadas, C. Caleiro, and T. Mossakowski. Categorial Fibring of Logics With Terms and Binding Operators. In *Frontiers of Combining Systems 2*, pages 295–316. Research Studies Press, 2000.

[124] A. Sernadas, C. Sernadas, and J. Valença. A Theory-Based, Topological Notion of Institution. In E. Astesiano, G. Reggio, and A. Tarlecki, editors,

Recent Trends in Data Type Specification, volume 906 of *Lecture Notes in Computer Science*, pages 420–436. Springer Verlag, 1995.

[125] A. Sernadas, C. Sernadas, and A. Zanardo. Fibring Modal First-Order Logics: Completness Preservation. *Logic Journal of the IGPL*, 10(4):413–451, 2002.

[126] J.-R. Shoenfield. *Handbook of Mathematical Logic*, chapter Axioms of Set Theory, pages 321–345. North Holland, 1977.

[127] M. Spivey. *The Z Notation: a Reference Manual.* Series in Computer Science. Prentice-Hall, 1992.

[128] Tardieu, Rochfeld, and Colletti. *La méthode MERISE, principes et outils (Tome 1 & 2).* Les éditions d'organisation, 1986.

[129] A. Tarlecki. On the Existence of Free Models in Abstract Algebraic Institutions. *Theoretical Computer Science*, 37(3):269–304, 1985.

[130] A. Tarlecki. Bits and Pieces of the Theory of Institutions. In D. Pitt, S. Abramsky, A. Poigné, and D. Rydeheard, editors, *Proceedings of the International Workshop on Category Theory and Computer Programming (IWCTCP'86)*, volume 240 of *Lecture Notes in Computer Science*, pages 334–363, 1986.

[131] A. Tarlecki. Quasi-Varieties in Abstract Algebraic Institutions. *Journal of Computer and System Science*, 1986.

[132] A. Tarlecki. Moving Between Logical Systems. In M. Haveraaen, O. Owe, and O.-J. Dahl, editors, *Recent Trends in Data Type Specification*, volume 1130 of *Lecture Notes in Computer Science*, pages 478–502. Springer Verlag, 1996.

[133] A. Tarlecki. Towards Heterogeneous Specifications. In D. Gabbay and M. de Rijke, editors, *Proceedings of Frontiers of Combining Systems (FroCoS'98)*, pages 337–360. Research Studies Press, 1998.

[134] A. Tarlecki. *Algebraic Foundations of Systems Specification*, chapter Institutions: An Abstract Framework for Formal Specifications. IFIP State-of-the-Art Reports. Springer Verlag, 1999.

[135] R. Thomason. *Handbook of Philosophical Logic*, volume 2, chapter Combinations of Tense and Modality. Kluwer Academic Publishers, 1984.

[136] W.M. Turski and T.S.E. Maibaum. *The Specification of Computer Programs.* Addison-Wesley, 1987.

[137] P. Veloso. On Pushout Consistency, Modularity and Interpolation for Logical Specifications. *Information Processing Letters*, 60(2):59–66, 1996.

[138] P. Veloso and S. Fiadeiro, J.-L.Veloso. On Local Modularity and Interpolation in Entailment Systems. *Information Processing Letter*, 82:203–211, 2002.

[139] P. Veloso and T. Maibaum. On the Modularization Theorem for Logical Specifications. *Information Processing Letter*, 53(5):287–293, 1995.

[140] P. Veloso and S. Veloso. On Local Modularity Variants and π-Institutions. *Information Processing Letter*, 77:247–253, 2001.

[141] M. Wirsing. Structures Algebraic Specifications: a Kernel Language. *Theoretical Computer Science*, 42:123–249, 1986.

[142] M. Wirsing. Structures Specifications: Syntax, Semantics and Proof calculus. In F.L. Bauer, W. Brauer, and H. Schwichtenberg, editors, *Logic and*

Algebra of Specification, volume 94 of *NATO ASI Series F*, pages 411–442. Springer Verlag, 1991.

[143] F. Wolter. A Couterexample in Tense Logic. *Notre Dame Journal of Logic*, 37(2):167–173, 1996.

[144] A. Zanardo, A. Sernadas, and C. Sernadas. Fibring: Completness preservation. *Journal of Symbolic Logic*, 66(1):414–439, 2001.