Cygne noir
Revue d'exploration sémiotique

Enjeux de la modélisation formelle en sémiotique computationnelle

Jean-Guy Meunier

Numéro 7, 2019

Algorithmes

URI : https://id.erudit.org/iderudit/1089329ar
DOI : https://doi.org/10.7202/1089329ar

Aller au sommaire du numéro

Éditeur(s)

Cygne noir

ISSN

1929-0896 (imprimé)
1929-090X (numérique)

Découvrir la revue

Citer cet article

Meunier, J.-G. (2019). Enjeux de la modélisation formelle en sémiotique computationnelle. *Cygne noir*, (7), 42–78. https://doi.org/10.7202/1089329ar

Résumé de l'article

Une sémiotique computationnelle n'est pas une sémiotique de la computation ni des outils numériques ou de la relation personne-machine. Il s'agit d'une approche qui dans sa modélisation des artefacts sémiotiques construit des modèles computationnels qui en appellent à des modèles conceptuels et formels. Plus précisément, un modèle formel se construit en regard de ce qui a été conceptualisé à propos des artefacts sémiotiques à l'étude et est aussi contraint par les exigences de la modélisation computationnelle. Cet article traite des enjeux de la modélisation formelle pour une sémiotique computationnelle, à savoir : (a) la modélisation formelle dans les pratiques sémiotiques, (b) les définitions d'une modélisation formelle, (c) le rôle déterminant d'un modèle conceptuel sur la modélisation formelle, (d) la contrainte qu'exerce la modélisation computationnelle sur la modélisation formelle et, enfin, (e) les limites qu'impose la modélisation formelle à la recherche sémiotique.

Enjeux de la modélisation formelle en sémiotique computationnelle

> [L]'analyse ne peut jamais se contenter de considérer les termes mais doit, par-delà les termes, saisir les relations qui les unissent. Ces relations seules constituent son véritable objet.
>
> Claude Lévi-Strauss[1]

1. Le défi de la sémiotique : la modélisation formelle

Vers la fin des années 2000, Peer Bundgaard et Frederik Stjernfelt ont entrepris une enquête au sujet des défis futurs de la sémiotique en interrogeant différents sémioticiens[2]. La conclusion était sans appel : la sémiotique se renouvellerait par la puissance du structuralisme dominant. On voyait ce paradigme comme une approche plus rigoureuse que celle offerte par la philosophie classique. Juri Lotman, bien auparavant, avait lui aussi insisté sur l'importance des mathématiques comme voie naturelle pour la formalisation de la sémiotique[3]. En 2014, Kalevi Kull et Ekaterina Velmezova ont repris cette enquête, en interrogeant un autre groupe de sémioticiens[4]. Cette seconde investigation donna cependant des résultats bien différents. On se montrait très pessimiste sur l'avenir de la sémiotique, car, selon ces chercheurs, le structuralisme n'avait pas livré la rigueur attendue. L'application du structuralisme en sémiotique avait peu exploré la formalisation de la notion de structure[5]. La sémiotique était demeurée surtout analytique et conceptuelle. Comme l'affirmait Mihai Nadin, la sémiotique était devenue trop « spéculative »[6]. Aujourd'hui, une problématique similaire émerge au sein de la communauté sémiotique qui, en raison de l'introduction de l'informatique dans sa pratique, s'interroge sur les enjeux de la formalisation dans une sémiotique qui devient computationnelle.

Dans cet article, j'explorerai ainsi les enjeux de cette modélisation formelle pour une sémiotique computationnelle. Je préciserai (a) les tentatives de modélisation formelle dans la pratique sémiotique ; (b) les définitions techniques d'une modélisation formelle ; (c) le rôle déterminant d'un modèle conceptuel sur la modélisation formelle ; (d) les contraintes qu'exerce la modélisation computationnelle sur la modélisation formelle ; et enfin (e) les limites qu'impose la modélisation formelle à la recherche sémiotique.

2. La modélisation formelle dans les pratiques sémiotiques

Malgré les critiques élaborées par les chercheurs interrogés par Kull et Velmezova, la recherche d'une rigueur formelle a toujours été présente en sémiotique. En effet, on peut trouver de nombreux exemples qui illustrent ces recherches formalisantes. Par exemple, Charles S. Peirce, comme on le sait, a identifié la sémiotique avec la logique. Il a utilisé des graphes existentiels[7]. Robert Marty a traduit ses catégories peirciennes en algèbre de treillis[8]. Louis Hjemslev représenta les relations entre la substance et la forme comme une relation fonctionnelle[9]. L'école sémiotique de Charles Morris utilisa la logique prédicative et modale de Frege-Carnap[10] qui ultérieurement fut surtout appliquée à l'étude de la langue et dont les formes les plus complexes furent offertes par la grammaire applicative de Sebastian Shaumyan et de Jean-Pierre Desclés[11]. Claude Lévi-Strauss utilisa des modèles algébriques[12] et Joachim et Michael Lambek, des grammaires catégorielles pour décrire des relations de parenté[13]. Avec des collègues, nous avons décrit des formes génériques iconiques par le biais de ces grammaires catégorielles[14]. A. J. Greimas proposa un carré logique pour sa structure actancielle[15]. Même Jacques Lacan et Roland Barthes proposèrent de « mathématiser » leurs objets d'étude[16] – la psychanalyse et la mode –, mais sans trop de succès. Jaakko Hintikka travailla la sémiotique dans l'horizon de la logique modale[17]. Jean Petitot, Line Brandt et Wolfgang Wildgen s'inspirèrent de la topologie et des systèmes dynamiques (à la Thom) pour expliquer la morphogénèse du sens[18]. Dans cette même voie, David Piotrowski et Yves-Marie Visetti proposèrent une modélisation diagrammatique pour penser une sémiolinguistique dynamique[19]. Enfin, des efforts pour une vision formelle se sont aussi retrouvés dans les théories sémiotiques naturalistes comme la biosémiotique[20], la cybersémiotique[21], la théorie de l'information[22] et de la communication[23] et la sémiotique cognitive[24]. Reste que la formalisation ne s'est pas constituée comme besoin de base de la pratique sémiotique. On la trouve réductionniste, positiviste, atomiciste et rarement dynamique. Évidemment, on doute assurément qu'une sémiotique computationnelle arrangerait les choses. Se pose alors la question de la valeur épistémique d'une sémiotique computationnelle dont les fondements reposent sur la modélisation formelle que l'on voit décidément comme réductionniste.

Pourtant, malgré toutes ces critiques, et contrairement à toute attente, la communauté des informaticiens et des mathématiciens, en dehors même de la communauté sémiotique, a commencé à appliquer systématiquement de la modélisation formelle à des objets sémiotiques par le biais de traductions algorithmiques. En effet, les sciences informatiques, surtout celles d'inspiration cognitive, ont exploré, avec une vision sémiotique, des territoires qui traditionnellement n'appartenaient pas aux sémioticiens.

Par exemple, certains chercheurs en ont appelé à la sémiotique pour comprendre des dimensions de l'informatique elle-même[25].

On les retrouve en grand nombre dans les sous-disciplines de la fouille d'information, de la fouille textuelle, du traitement du langage ou du traitement d'images, etc. Des logiciels de divers types, souvent clé en main, sont utilisés pour effectuer certaines tâches d'analyse sémiotique. Cachés dans ces algorithmes et ces logiciels, se trouvent des grammaires formelles de tout type servant au traitement du langage, ou encore de l'algèbre linéaire et non linéaire pour le traitement d'images, de la parole ou de la musique, et enfin des modèles chaotiques, dynamiques, catastrophiques, évolutionnistes pour le traitement de la cognition. Quelques rares mathématiciens et informaticiens (à l'exemple de James Albus, Alexander Meystel, Jean Petitot, Ricardo Gudwin, Fernando Gomide, Burghart Rieger, Kumiko Tanaka-Ishii, Leonid Perlovsky, notamment)[26] verront explicitement un travail de nature sémiotique dans ces algorithmes et ces modèles formels.

Qui plus est, l'informatique contemporaine (l'intelligence artificielle, la fouille de données, l'apprentissage machine, l'apprentissage profond, etc.) a commencé à explorer de manière computationnelle des artefacts sémiotiques très classiques comme le langage naturel, les arts visuels, la musique, les jeux, les rituels, les médias, pour ne citer que cela. Là encore, on y traite de manière originale des données sémiotiques. Bien que ne quittant pas les approches classiques déductives descendantes (*top down*), elles introduisent maintenant des méthodes dites ascendantes ou inductives qui permettent la découverte de régularités structurales à propos des relations entre les données et donc ultimement du sens et de la signification de ces données.

Enfin un courant important issu de ces approches informatiques en est même émergé à la fin des années 1980 : les *humanités numériques* (HN)[27]. Celles-ci ont de fait exploré des territoires appartenant classiquement à la sémiotique tels que le texte, le discours, l'argumentation, le genre, le style, les contenus iconiques, la musique, etc. Cependant, dans ces HN, l'informatique était souvent utilisée comme un outil qui, soutient-on, n'affectait pas la pratique classique des sciences humaines et sociales et des arts et des lettres : « Les Humanités numériques sont une extension des compétences et des méthodes traditionnelles du savoir, elles ne les remplacent pas[28]. »

Pourtant, malgré la richesse des recherches menées dans le domaine, il semble bien que ces HN n'aient pas influencé de façon importante les sémioticiens – à l'exception de rares chercheurs en sciences humaines aux intérêts sémiotiques avoués qui ont intégré l'informatique dans leur démarche[29]. La modélisation formelle de dimensions sémiotiques est maintenant entrée dans les projets des HN. Elle y demeure cependant peu explicitée. Le formel reste caché sous les algorithmes ou sous la technologie informatique qui est souvent utilisée comme un outil dont on ignore les fondements formels[30].

Le courant semble aujourd'hui changer. Pour une nouvelle génération de jeunes sémioticiens et sémioticiennes, la sémiotique future doit non seulement utiliser ces outils informatiques, mais elle doit aussi comprendre et s'approprier la modélisation formelle et computationnelle qui lui est sous-jacente. Ce n'est qu'à cette condition qu'elle pourra l'appliquer avec rigueur et de manière heuristique à des artefacts sémiotiques. Ces modélisations formelles et computationnelles ouvriront de nouvelles voies de recherche pour la sémiotique. Si la communauté sémiotique ne s'approprie pas ces modélisations, les recherches sur les artefacts sémiotiques n'en continueront pas moins, mais il est fort possible qu'une bonne partie de celles-ci soient réalisées dans des sciences aux méthodes de type naturaliste, positiviste ou expérimental. N'est-ce pas ce que font déjà diverses disciplines des sciences cognitives, comme la psychologie, l'intelligence artificielle, la sociosémantique, les technologies de l'information et les études en communication? Les questions sémiotiques comme telles ne disparaîtront pas, mais les analyses sémiotiques demeureront conceptuelles et la philosophie y jouera un rôle d'épistémologie des théories sur le signe.

Par ailleurs il semble que la sémiotique est déjà confrontée aux problèmes que pose l'usage de l'informatique. De fait, elle en vit le défi sous divers couverts. Et, comme c'est le cas pour plusieurs autres disciplines scientifiques, l'informatique pourrait entraîner une modification profonde des objets d'étude de la sémiotique et de sa méthodologie[31]. On peut penser par exemple à l'impact des données massives (*Big data*) sur la sémiotique. Propp et Lévi-Strauss avaient réussi à étudier quelques dizaines de contes ou de légendes. Il leur serait maintenant possible d'en explorer des milliers. Et un tel nombre de données introduit nécessairement dans la recherche une note empirique plus forte. Ces données ne changent pas l'artefact sémiotique comme tel. Elles demeurent toujours des données « signifiantes ». Elles sont simplement en plus grand nombre. Cependant ce qui change profondément est notre vision de ces objets d'étude. Et une modélisation computationnelle invite nécessairement une modification de la méthode d'investigation et d'analyse.

Cependant, si une telle vision computationnelle de la sémiotique veut se développer en regard de l'informatique, elle doit mieux comprendre ses fondements théoriques et mieux les positionner en regard d'une formalisation ; formalisation qui devient une condition *sine qua non* de son entrée dans l'univers computationnel et donc de l'utilisation de l'ordinateur.

3. La modélisation conceptuelle dans une sémiotique computationnelle

Souvent, on comprendra la formalisation comme une opération effectuée dans un « troisième monde » où elle est complètement détachée du monde réel ou actuel. Or comme le montre la philosophie récente des sciences[32], la formalisation est une opération épistémique qui ne peut pas être détachée ou isolée de multiples autres opérations cognitives à l'œuvre dans une démarche scientifique et plus particulièrement de la modélisation conceptuelle.

Ceci est d'autant plus vrai dans le cas d'une modélisation formelle qui serait réalisée au sein d'une recherche sémiotique. En effet, il ne peut exister de modèles formels sémiotiques qui soient détachés de l'important travail conceptuel qu'une recherche sémiotique met en œuvre. Ce travail conceptuel est inhérent à la sémiotique en raison de la complexité des objets d'étude. Ceux-ci sont la complexité incarnée. Ils ne révéleront pas facilement leur signification à l'analyse, car celle-ci se tisse dans de multiples relations des plus intriquées. Pour paraphraser Peter Frederick Strawson, les signes ne portent pas leur signification sur leurs manches[33].

Cette modélisation conceptuelle est un travail omniprésent dans les recherches sémiotiques. Je dirais que le modèle conceptuel est celui qui consume le plus d'énergie épistémique. Et il ne doit pas être négligé. Tout comme les autres démarches scientifiques, la sémiotique doit conceptualiser ses propres objets d'étude pour révéler les constituants des artefacts sémiotiques et les relations structurantes qui les caractérisent. On ne peut pas penser que de tels objets complexes puissent être modélisés formellement sans, au préalable, avoir été conceptualisés d'une manière ou d'une autre. Comme le dit l'épistémologie classique, l'objet d'étude doit être *appréhendé*, c'est-à-dire non seulement senti, mais *conceptualisé*. Cette conceptualisation ne reste pas purement mentale pour un individu. Elle doit aussi en arriver à être exprimée dans un langage naturel pour en permettre la communication à une communauté épistémique. Ce travail de conceptualisation et d'expression linguistique est tantôt préalable tantôt concomitant à un éventuel modèle formel ou computationnel[34].

Comme le dit autrement Gilles-Gaston Granger, une modélisation formelle n'est qu'un horizon dans l'ensemble d'une démarche scientifique :

> En tant que pensée en exercice, elle [la science] ne peut se présenter que comme une tentative de mise en forme, commentée par le truchement d'un langage non formel. La formalisation totale n'apparaît jamais que comme *un horizon de la pensée scientifique*, et l'on peut dire que la collaboration des deux langages est un caractère transcendantal, c'est-à-dire dépendant des conditions mêmes d'appréhension d'un objet[35].

Illustrons la nécessité d'un tel modèle conceptuel dans une analyse sémiotique. Commençons par un premier exemple simple : la sémiotique d'une fête d'anniversaire d'enfants dont on veut expliciter les constituants et les relations sémiotiques. Une analyse sémiotique de ce rituel ne peut pas débuter en projetant sur ce rituel un modèle formel, par ex. : un graphe des actions, une équation sur la dynamique d'échanges d'objets, etc. Il faut au préalable conceptualiser ce qui constitue ce rituel et exprimer ces concepts dans un langage naturel. Et selon les paradigmes théoriques dans lesquels le sémioticien se situera, il en appellera à diverses expressions conceptuelles pour désigner ces entités et ces relations. Certaines entités seront conceptualisées et catégorisées comme *acteur, actant, sujet, objets, instruments, lieu, temps*, etc. Elles se verront attribuer des propriétés, des traits, des caractéristiques, des valeurs, des désirs, des émotions, etc. On parlera de certaines relations en termes d'actions, de praxis, d'opérations, d'opposition, de dépendance, de proximité, d'association, de destination, d'orientation, d'intention, de comportements, de rituels, etc. On en appellera aussi à un métalangage théorique où des entités et des relations seront décrites comme des représentations, des désignations, des observations, de la signification, de l'interprétation, du sens, de la référence, des énonciations, des discours, des images, du culturel, du social, etc. Ces termes et les énoncés qui les contiennent varieront selon les paradigmes. Mais, quelles qu'en soient la liste et les formes, ces concepts et ces catégories relèvent tous, non pas du modèle formel et computationnel comme tel, mais bien d'un modèle conceptuel. Or c'est souvent ce modèle conceptuel qui est, comme je le disais précédemment, le travail central qui occupe une grande partie de l'énergie de la recherche sémiotique. Souvent transparent aux chercheurs eux-mêmes, il n'en demeure pas moins présent. C'est donc dans le modèle conceptuel qu'un modèle formel puise ultimement les référents et le sens de ses symboles et formules. En d'autres termes, c'est dans le modèle conceptuel que la sémantique des symboles et des formules est donnée. Encore une fois, sans lui, il n'y a pas de compréhension.

Voici un second exemple où la complexité de l'objet est plus évidente : la définition du signe. Encore une fois, l'objet d'étude doit être conceptualisé avant même d'être formalisé. Pour Saussure, le signe est conceptualisé *comme une relation entre le signifiant et le signifié*. Pour Peirce, il est *une structure relationnelle entre le representamen, l'objet, l'interprétant* et même *l'interprète.* Pour Frege, il est *une structure relationnelle entre un signe (zeichen) ou expression (ausdruck), un sens (sinn) et une référence (Bedeutung)*, notamment. Dans de multiples sciences cognitives, un signe (symbole) est conceptualisé comme un ensemble complexe de relations entre des facteurs *subjectifs, perceptuels, catégorisants, conscients, sociaux, culturels, contextuels,* etc.

Autrement dit, on conceptualise un artefact sémiotique comme un « objet » complexe structuré de plusieurs relations. Quel que soit le modèle formel que l'on pourrait projeter sur lui, sa sémantique référera à ce qui est construit dans le modèle conceptuel. Ainsi avant même de penser à de la formalisation mathématique pour décrire ces relations structurantes, un grand travail de modélisation conceptuel est nécessaire.

Dans la perspective d'une sémiotique computationnelle, une telle modélisation conceptuelle est essentielle. Il est illusoire de croire que des outils informatiques peuvent remplacer ce travail. Un modèle conceptuel est la base à partir de laquelle les concepts et les énoncés peuvent exprimer de manière compréhensible aux humains les relations structurantes à l'œuvre dans la sémioticité des artefacts étudiés. Ce seront justement ces conceptualisations des relations dans lesquelles les modèles formels et computationnels puiseront pour leur propre construction. Plus encore, ces modélisations conceptuelles seront ultimement l'horizon qui permettra l'interprétation des résultats de la formalisation et de la computation, même si celles-ci découvrent des relations nouvelles, passées inaperçues au chercheur au début de sa recherche.

En somme, en sémiotique, même si la modélisation conceptuelle est imprécise, générale, subjective, biaisée, influencée, etc., et même si le langage naturel dans lequel elle s'exprime est ambigu, métaphorique, fictionnel, narratif, inconsistant, etc., elle n'en demeure pas moins un moment épistémique incontournable dans l'explication et la compréhension du sens et de la signification des artefacts signifiants qui sont ses objets d'étude. Elle offre un premier dévoilement des relations structurantes qui sous-tendent ces types d'artefacts. Ces relations et leur relata seront souvent identifiées comme étant des propriétés, *des objets, des évènements*, des *traits, des caractéristiques, des états*. Ce sont ces relations auxquelles référeront, certes de manière complexe, les symboles des modèles formels et qui deviendront *des variables, des constantes, des opérations*, etc. Et parce que ces relations structurantes sont souvent très nombreuses, mais surtout complexes, il devient nécessaire de construire une modélisation formelle réglée, contrôlée et qui s'exprime dans un langage non ambigu. Ce modèle formel n'aura donc ultimement de signification qu'en relation à ces relations structurantes conceptualisées dans le modèle conceptuel. Un modèle formel ne se comprend qu'en relation à ce modèle conceptuel. Comme le dit très justement Pierre Duhem, donner à un physicien une équation, sans lui donner d'explication, il n'y comprendra rien[36].

Ainsi, quelle que soit la démarche scientifique effectuée, un modèle formel est contraint par un modèle conceptuel. Ce dernier est l'horizon épistémique qui présente ce qui est à modéliser formellement. Il est aussi l'horizon à partir duquel les résultats de la manipulation formelle seront interprétés. Dans une telle perspective, un modèle conceptuel est une contrainte majeure auquel un modèle formel est soumis. Il ne s'agit cependant pas de la seule contrainte. En effet, dans le cas d'une relation possible avec l'informatique, ce ne sont pas toutes les relations structurantes identifiées dans la modélisation conceptuelle et retenues par une modélisation formelle qui pourront être intégrées dans une modélisation computationnelle. Un modèle conceptuel exerce donc une contrainte importante sur un modèle formel, mais ce ne sera pas la seule contrainte que rencontre la modélisation formelle. Le modèle formel est aussi contraint en aval par le modèle computationnel auquel il sera associé. Dit autrement, ce n'est pas parce qu'un modèle formel est créé qu'il est immédiatement admissible à la computation! En cela la différence entre la perspective de la sémiotique structurale et la sémiotique computationnelle est immense. Cependant, avant d'expliquer cette contrainte de la computationnalité sur la formalisation, rappelons plus précisément, mais brièvement, les définitions techniques d'un modèle formel.

4. Définitions techniques d'une modélisation formelle

En philosophie des sciences et en épistémologie, le modèle formel a été considéré, de façon classique, comme le modèle idéal et prototypique de la science : pas de modèle formel, pas de science rigoureuse. Comme le résume bien Teodor Shanin :

> Les disciplines savantes incorporent les éléments de formalisation. [...] D'une part, cela reflète le besoin de l'esprit humain de rationaliser et de s'orienter dans une réalité immensément complexe, d'y parvenir en construisant des images générales systématiques et cohérentes et des schèmes symboliques[37].

Et dans l'horizon d'une sémiotique qui se veut computationnelle, ce type de modèle est une condition *sine qua non* d'une modélisation computationnelle et ultimement de l'utilisation de la technologie informatique. Il est donc important de comprendre ce qu'il en est de ces modèles formels dans le cadre d'une sémiotique computationnelle.

Dans cette perspective, un modèle formel est un artefact sémiotique qui se définit classiquement sur trois plans interreliés : (1) un plan syntaxique, (2) un plan sémantique et (3) un plan pragmatique.

4.1 Définition syntaxique d'un modèle formel

Au plan syntaxique, un modèle contient nécessairement une structure sémiotique ou des expressions formées à partir d'un ensemble de symboles dont la manipulation est régie par un ensemble de règles. Selon l'expression de Jean Ladrière : « Un formalisme se présente comme un système de symboles soumis à des règles précises de manipulation[38]. » On retrouve une même définition simplifiée chez les scientifiques comme David Marr : « Dire que quelque chose est un modèle formel signifie seulement qu'il s'agit d'un ensemble de symboles avec des règles pour les assembler – ni plus ni moins[39]. »

Certaines disciplines universitaires appelleront un tel artefact sémiotique un *langage formel* ou un *système symbolique formel.* Dans ce cas, on dira que ce langage est constitué d'un ensemble de symboles[40] appelés le *vocabulaire* ou le *lexique* et d'un ensemble de règles définissant la *syntaxe ou la grammaire* de ce langage. Selon le langage formel choisi, il existera diverses manières de représenter la diagrammatique, c'est-à-dire la manipulation formelle des symboles et de leur composition, mais surtout de leurs transformations.

Un même système formel peut cependant être présenté dans les formes symboliques différentes et qui sont régies par des grammaires qui leur sont propres. On parlera alors de *formalisme* ou de *notation*. Par exemple, le langage formel qui exprime une logique des propositions pourrait être un formalisme russellien ou un formalisme polonais[41]. Les deux pourront exprimer les mêmes structures logiques et ils pourraient construire des preuves équivalentes. Et selon les types de notations, la diagrammatique représentera des séquences de manipulations des symboles : *composition, dérivation*, *génération*, *tableau*, *arbre* et même de *calcul*, *raisonnement*, etc. Ce sont ces séquences de manipulations réglées qui entreront dans une *démonstration*, une *preuve*, etc.

Sur le plan épistémique, tous les formalismes ou notations ne sont pas équivalents. Tous n'expriment pas nécessairement les mêmes dimensions des structures logiques ou mathématiques. Par exemple, ils n'expriment pas tous les mêmes types ou catégories des symboles, formules et règles. Surtout, ils n'offrent pas toute une même facilité de manipulation et de compréhension pour les humains.

Enfin, dans certains cas, on assignera des statuts particuliers à certaines formules de ce langage (formules, équations, etc.). Par exemple, elles pourraient être prises comme des axiomes, des postulats, des théorèmes ou des corollaires. Et si, en plus, toutes les formules d'un langage peuvent être construites par divers types de règles strictes de composition, d'inférence, de transformation, etc., on parlera alors d'une formalisation de type axiomatique. Le prototype classique d'un système axiomatique est celui de la logique des propositions, mais plusieurs autres systèmes formels existent quoique leur structure axiomatique ne soit pas toujours transparente à l'utilisateur.

On ne doit pas confondre un *langage* axiomatique avec un *modèle* mathématique. L'un n'est pas l'autre. Un langage axiomatique est une forme particulière de système formel alors qu'un modèle mathématique est un artefact épistémique qui utilise un ou plusieurs types de langages formels capables d'exprimer certaines structures mathématiques. Autrement dit, une même structure mathématique pourrait être exprimée dans des langages formels différents. Par exemple, certaines structures mathématiques pourraient être exprimées dans un formalisme algébrique, un formalisme géométrique ou un formalisme topologique, etc. Cependant tous les langages formels ne se prêtent pas à l'expression d'une même structure mathématique. Un langage formel est souvent formé en regard de la sémantique qui lui sera associée et du rôle pragmatique qui lui sera assigné. Par exemple, bien qu'il y ait une même structure mathématique pour calculer la surface d'un jardin de forme carrée, il y aura des différences importantes entre les langages pour exprimer cette même structure mathématique. Ainsi, les langages de l'arithmétique, de l'algèbre, de la géométrie, de la topologie et les treillis n'ont pas tous les mêmes facilités de manipulations des symboles pour les humains. Pourtant, ces langages arriveront à des résultants *équivalents*.

4.2 Définition sémantique d'un modèle formel

Techniquement parlant, le langage formel ne possède aucune sémantique. Il est un système de symboles sans signification. Or, un modèle ne peut pas n'être qu'un langage formel, si bien formé soit-il. Comme tous les autres modèles construits dans une recherche, un modèle formel doit être mis en correspondance avec quelque chose d'autre. En termes classiques, un modèle doit « représenter » quelque chose. Pour le dire autrement, il faut ajouter une sémantique à un langage formel. Comme le rappellent Mauricio Suárez et Albert Solé : « Dans une représentation scientifique, une source A – habituellement un modèle, un graphe, une équation – est employée pour représenter une cible B – habituellement un système, une entité ou un phénomène[42]. »

La nature spécifique de la sémantique des langages formels ne va évidemment pas sans générer des débats. Les Peirce, Frege, Russell, Husserl, Carnap, Tarski, Kripke et Bourbaki de la communauté des mathématiciens, par exemple, ne définissent pas la sémantique des langages formels de la même manière. Cependant, malgré des différences majeures entre ces sémanticiens du formel, il en ressort un point commun. Quel que soit le langage formel utilisé, une sémantique doit déterminer une référence : elle doit mettre le langage formel en correspondance avec autre chose. De fait, la thèse sémantique dominante dans ces débats sera de type référentiel. On ne nie pas le problème du « sens », mais on donne priorité à la référence. Dans cette perspective, les divers types d'expressions construites dans ce langage formel, comme les formules, les équations, les graphes, etc., ne sont vrais ou faux que relativement à cette sémantique. On dit alors qu'un tel langage formel est *interprété*.

La question qui est alors soulevée par cette thèse est la suivante : à quoi alors réfère le langage formel le plus communément utilisé dans les modèles scientifiques, à savoir le langage mathématique? Une des réponses classiques à cette question soutiendra que les mathématiques *réfèrent*[43] à des nombres. Ce que l'on interprétera souvent de manière rapide en identifiant une sémantique mathématique à une sémantique de la quantité. Or, la majorité des mathématiciens et mathématiciennes refuseront de réduire la sémantique des langages mathématiques à une référence à des nombres[44]. Ce sera d'ailleurs l'une des grandes contributions de l'école mathématique de Bourbaki de montrer que la sémantique des langages mathématiques ne se réduit pas à l'univers des nombres. Au contraire, pour Bourbaki, l'essentiel de la mathématique est avant tout de porter sur des « structures », c'est-à-dire sur la diversité de relations qui peuvent exister entre des éléments, peu importe le monde dans lequel ils peuvent être instanciés. L'univers de nombre n'étant qu'un possible parmi d'autres.

Inspiré par les analyses de Michael S. Mahoney en histoire des mathématiques, je dirais aussi que les mathématiques ne portent pas sur le monde comme tel, mais sur la manière dont on se représente le monde et plus particulièrement sur la manière dont on se représente les relations que nous construisons à propos de ce monde, quel qu'il soit[45]. Autrement dit, ce dont parlent les mathématiques (leur sémantique) est à propos de certains types particuliers de structures de relations entre des éléments, ce que Bourbaki appelle des *structures mathématiques*[46].

4.3 Définition pragmatique d'un modèle formel

La troisième dimension d'une modélisation formelle est de nature pragmatique. Celle-ci renvoie au rôle épistémique d'une modélisation formelle dans une démarche scientifique.

Une sémantique, si spécifique soit-elle, n'émerge pas du langage lui-même. Un langage formel ne génère pas sa sémantique de lui-même[47]. Celle-ci vient plutôt des créateurs du modèle qui utilisent ce langage dans une modélisation. Ainsi, le choix d'une sémantique dépendra en partie des intentions épistémiques des créateurs du modèle[48] lorsqu'ils construisent un langage formel pour représenter leur objet d'étude.

À titre d'exemple, prenons un objet physique comme le pendule, que des chercheurs voudraient modéliser formellement. Pour ce faire, ils auront à choisir une modélisation spécifique parmi un grand nombre de modèles formels possibles. L'un d'eux pourrait servir à formaliser uniquement la *période de temps* qu'un pendule prend pour une oscillation. On pourrait ainsi vouloir modéliser *la friction* du mécanisme de suspension ou encore la *résistance du matériau* de la tige relativement à la masse. Et, pour chacun de ces « angles de vues » du pendule, une formalisation exprimant une structure mathématique spécifique pourrait être utilisée. Et pour chacune de ces structures mathématiques, plusieurs langages mathématiques sont disponibles. L'un d'eux pourrait être algébrique, l'autre topologique, etc.

Sur le plan pragmatique, ce modèle formel a un rôle de médiateur épistémique[49]. Il sert à « représenter » *sous un angle ou un autre* son objet d'étude. Dans un contexte scientifique, il faut cependant bien comprendre que le terme « représentation » n'a pas la signification métaphorique que le discours ordinaire lui donne. En effet, un modèle formel ne « représente » pas le réel comme un miroir de la réalité. Il ne reflète pas la réalité. Il le rend simplement « présent » au chercheur sous une autre forme.

Un modèle formel est ainsi un outil épistémique, c'est-à-dire une construction cognitive qui permet à des chercheurs de mettre en correspondance des équations ou des formules d'un langage formel avec l'objet d'étude. De fait, le modèle *projette* des structures mathématiques sur ses objets d'étude. Pour parler encore métaphoriquement, un modèle est un filtre appliqué à l'objet d'étude et non un miroir réfléchissant. Il projette ainsi des schémas structurants sur l'objet d'étude. Comme l'affirme Giorgio Israel : « les schémas mathématiques unifient des réalités différentes, mais isomorphes[50]. »

Cette projection elle-même peut être traduite formellement comme *une application* (*mapping*) fonctionnelle qui va d'une structure mathématique (SM) particulière à une structure spécifique de l'objet (SO) (fig. 1).

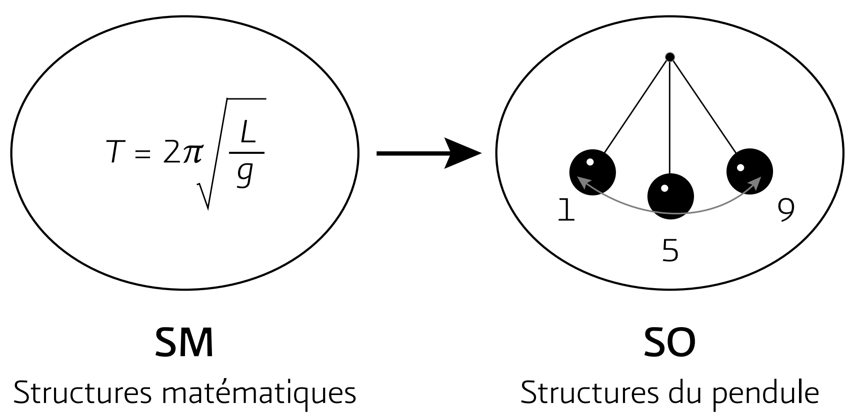

Figure 1. La modélisation formelle du pendule comme projection.

Cette formalisation tant des objets d'étude que de la relation de modélisation elle-même devient des plus intéressantes en science. Lorsque les objets d'étude sont complexes et dynamiques, elle permet d'unifier dans un langage formel commun une multitude de propriétés structurales de l'objet, mais aussi de comparer les modèles eux-mêmes. Ce type de formalisation ouvre alors à des analyses « différentielles » et « distributionnelles ». Elle fait ainsi apparaître de nouveaux « observables », c'est-à-dire qu'elle révèle de nouvelles propriétés de l'objet par la manipulation des symboles qui « représentent » ces propriétés dans le modèle formel lui-même.

La définition générale des modèles formels que j'ai présentée sous les angles syntaxique, sémantique et pragmatique est relativement classique. Je souhaite cependant souligner un point important de la sémantique : celle suggérée est d'inspiration bourbakienne. Elle lie directement le langage mathématique à des structures mathématiques. Or, cette sémantique est une des voies par lesquelles il devient possible de lier les langages formels mathématiques à la modélisation computationnelle pertinente à une sémiotique computationnelle. Dans cette définition une modélisation formelle ne peut être réduite à n'être qu'une modélisation quantitative.

Ainsi, si l'un des intérêts de la sémiotique computationnelle est d'étudier les propriétés signifiantes des artefacts porteurs de signification, il lui faut les approcher par une modélisation formelle dont la sémantique doit référer à des structures caractérisant ces artefacts, que celles-ci soient discontinues ou continues, stables ou dynamiques. Ce sont elles qu'un modèle formel projettera comme un filtre. Mais, comme nous le verrons dans ce qui suit, dans le cadre de l'intégration de l'informatique, ce modèle sera soumis à une autre contrainte : celle de la computation.

5. Modélisation formelle et computation

Il existe évidemment un grand nombre de structures de dépendances fonctionnelles qui peuvent être exprimées dans un langage formel. Cependant, dans un contexte informatique, il en est une sorte particulière qui est essentielle et nécessaire pour construire une modélisation computationnelle, à savoir celle qui, précisément, possède une structure de relations dites de type fonctionnel. En fait, ces types de relations forment un sous-ensemble de toutes les relations possibles entre des éléments. Elles sont définies techniquement de la manière suivante : une relation est dite fonctionnelle, si les éléments x_i d'un ensemble de départ X déterminent ou sont associés à un seul et unique élément y_i dans l'ensemble d'arrivée Y (fig. 2).

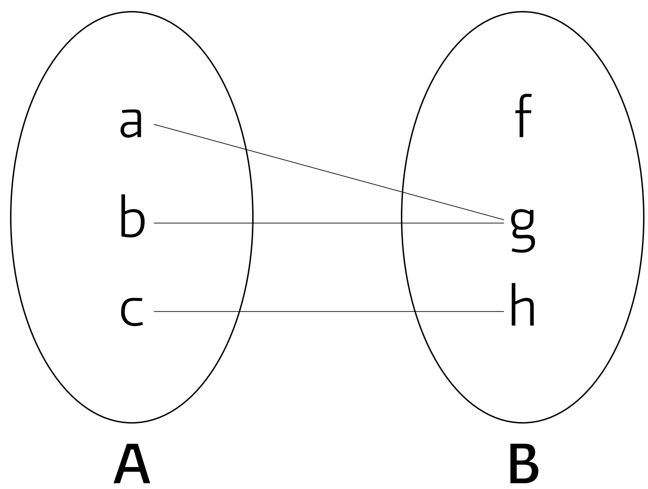

Figure 2. Graphe d'une relation fonctionnelle entre deux ensembles.

Autrement dit, une relation est dite fonctionnelle (ou fonction) à partir du moment où un élément de l'ensemble de départ ne peut pas avoir une relation avec plusieurs éléments de l'ensemble d'arrivée.

Pour illustrer ce type particulier de relations, nous prendrons deux exemples, l'un est arithmétique et l'autre non arithmétique.

L'exemple arithmétique est la relation fonctionnelle appelée le « carré d'un nombre ». Cette fonction détermine que pour tout nombre *X* d'un premier ensemble, il ne peut y avoir qu'un nombre *Y* dans l'ensemble d'arrivée. On peut représenter cette fonction de plusieurs manières. En algèbre, on écrira l'équation suivante :

$$y = x^2 \text{ ou } F(x) = x \cdot x$$

Cette manière de présenter une fonction est dite « intensionnelle ». Elle pourrait également être écrite dans le formalisme de la théorie des ensembles sous forme d'une liste de doublets :

$$F\ (x, y) : \{(1,1), (2,4), (3,9)\ldots (n, n)\}$$

Cette manière de présenter la même fonction sous la forme d'une liste est dite « extensionnelle ».

Notre deuxième exemple est non arithmétique. Il montre que les relations fonctionnelles n'existent pas uniquement dans l'univers des nombres. Ainsi, les sémanticiens considéreront que l'expression linguistique « Fils de » exprime une relation fonctionnelle entre deux personnes. En effet, cette relation détermine que, pour tout *X* qui est le fils de *Y*, il n'y a qu'un seul *Y* auquel *X* est lié. La langue française appellera par exemple ce *Y* « père ». Tout comme dans le cas précédent, il faut une procédure effective quelconque pour déterminer quelle est effectivement la personne *Y* qui est associée à la relation « X est le fils de ». Comme dans le premier exemple, cette relation fonctionnelle peut être exprimée « intensionnellement » ou « extensionnellement ».

Intensionnellement, elle sera formulée dans un langage logique prédicatif comme $\forall x\ \forall y$ (FILS-de $[x_i, y_i]$). En langage naturel, cette expression logique sera traduite par la phrase : *une personne est le fils d'une autre personne.* Extensionnellement, elle sera formulée par une liste de doublets : {(Jean Tremblay, Pierre Tremblay), (Louis Beaupré, Pierre Beaupré)… (N_1, N_2)}.

Ces types de relations fonctionnelles se retrouvent régulièrement exprimées dans les langages mathématiques. En fait, elles forment un sous-ensemble particulier des structures mathématiques dont je parlais plus haut et que les langages mathématiques ont pour fonction d'exprimer. En contexte informatique, ces fonctions ont un rôle encore plus important : elles deviennent l'une des conditions *sine qua non* de leur computationnalité. Voyons pourquoi.

Reprenons notre exemple arithmétique de la fonction « mise au carré ». Pour trouver la valeur de cette fonction « $F(y) = x \cdot x$ », il faut effectuer une opération que l'on appelle de manière générale un *calcul*. En l'occurrence, cette opération consiste à *multiplier une valeur de* x_i *par lui-même*. Or cette opération de « multiplication de x_i par lui-même » n'est pas simple : elle exige une manipulation sophistiquée de symboles[51]. On sait combien ces opérations exigent des efforts d'apprentissage chez les jeunes enfants.

Ces exemples montrent cependant quelque chose de très important : même si une expression formelle est bien formée, il ne s'ensuit pas que l'on connaît ou même qu'il existe une « procédure effective » pour trouver pour tous les éléments d'un ensemble quel est l'autre élément auquel il est associé. Dit autrement, il ne faut pas confondre l'expression d'une relation fonctionnelle avec les procédures de calcul effectif. Ce n'est pas parce qu'on a une équation formelle exprimant une relation qu'il existe une procédure effective de « calcul » qui donnera pour chaque valeur x_i une valeur y_i. Plus techniquement, on dit que si pour des relations fonctionnelles données il existe des « procédures effectives » pour trouver pour toute valeur de x une certaine valeur de y, alors cette fonction sera dite *calculable* (en anglais : *computable*).

Existe-t-il toujours de telles procédures de calcul? Est-ce que toutes les fonctions construites en mathématiques sont calculables? Ce sont d'importantes questions posées par David Hilbert au début du XIXe siècle à propos de ce « calcul »[52]. Comment peut-on prouver qu'il existe ou n'existe pas une telle procédure pour toutes les relations fonctionnelles mathématiques? Autrement dit : qu'est-ce qu'une fonction calculable? Ou encore : qu'est-ce qui garantit qu'une fonction est calculable?

Deux voies de réponses ont balisé la compréhension de cette notion de calculabilité des fonctions et ont permis de la relier à la notion qui lui sera associée. Une première creusa la notion de calculabilité elle-même, l'autre explora la notion de structures mathématiques. Les deux démontrèrent des convergences théoriques importantes sur l'équivalence entre certaines formes de calculabilité et certaines structures mathématiques. Cette équivalence sera importante pour comprendre les liens de complémentarité sinon même d'équivalence entre certains modèles computationnels et certains modèles formels. J'y insisterai donc un peu.

La première voie qui explora ce lien de la calculabilité des fonctions avec la computationnalité commença avec Alan Turing en 1936[53]. Une de ses grandes contributions fut, comme le souligna Gödel, de montrer que ces notions de décidabilité et de calculabilité pouvaient être comprises comme un ensemble de *procédures effectives* de manipulation de symboles représentant une fonction mathématique (appelée par Turing une *machine abstraite*) et qui seraient réalisables par un type particulier de *machine physique* (appelée plus tard « Machine de Turing »). On dira alors que si la fonction était « calculable », elle était aussi « computable » par cette machine.

Plus tard, une avancée théorique importante, appelée « thèse Church-Turing », affirma que les fonctions computables par une machine de Turing étaient elles aussi computables par une autre forme de calcul : le lambda-calcul de Church[54]. Cette thèse fut ultérieurement généralisée à plusieurs autres formes de computation, par exemple : la logique combinatoire[55], les algorithmes[56], les règles de production[57] et bien d'autres. Qui plus est Robin Gandy démontra que la computation pouvait aussi être parallèle et dynamique[58].

Il suivra de toutes ces recherches la généralisation suivante : il y a des équivalences entre des relations fonctionnelles calculables et diverses formes de computation comme les algorithmes, les règles de production, les automates, certaines grammaires de Chomsky, etc.

La deuxième voie de recherche pour lier calculabilité et computationnalité sera parallèle à cette dernière et en reprendra certains jalons. Un peu plus complexe dans son argumentation, elle fera un lien entre les structures mathématiques et certaines formes de computation. Plusieurs mathématiciens, par des arguments formels différents et sophistiqués, montreront que certaines de ces structures mathématiques, plus particulièrement les structures algébriques, lorsque comprises dans leur formulation d'algèbre abstraite, peuvent être mises en lien avec des fonctions computables. Par exemple, des formules algébriques construites par diverses sortes de *combinaisons ou transformations ou compositions* réglées sont vues comme des groupes, des sous-groupes, des monoïdes, des anneaux, etc. Von Newman montra que ces structures abstraites pouvaient être mises en lien avec les théories combinatoires et le lambda-calcul de Church. Enfin, il reliera ces structures aux *automates* qu'il avait lui-même développés : « La théorie des automates [...] devra être, d'un point de vue mathématique, combinatoire plutôt qu'analytique[59]. »

Chomsky reliera à son tour ces automates aux structures d'un langage[60]. Sa théorie montra qu'il était possible de faire une équivalence entre la théorie des « automates » finis de Von Neuman et des types de grammaires génératives[61]. De cette manière, certaines syntaxes formelles devenaient équivalentes à des automates finis déterministes : « La grammaire d'une langue peut être considérée comme une théorie de la structure de cette langue[62]. » Cette thèse chomskienne sera reprise par Marcel-Paul Schützenberger qui, lui, la remettra en lien avec la théorie des algèbres abstraites[63]. Il montrera que certaines de ces structures pouvaient exprimer abstraitement la nature mathématique de la formation de séquences de symboles dans des langages. Il y avait donc un lien profond entre des automates, certains types de grammaires structurales et des structures mathématiques abstraites[64]. Daniel Lacombe pour sa part montrera que des fonctions sur du continu était aussi computables[65].

Il manquait cependant un pont avec l'informatique classique issue de Turing et Church. Ce lien fut développé par John McCarthy[66]. Vers les années 1960, celui-ci créa le célèbre langage de programmation Lisp qui lui-même avait un lien profond avec le lambda-calcul de Church. McCarthy montra alors qu'on pouvait comprendre des automates et des algorithmes comme des opérateurs de transformation d'expressions symboliques tels qu'ils sont construits par des grammaires formelles de Chomsky. Finalement, Christopher Strachey et d'autres proposèrent de se servir de ces types de structures comme fondement mathématique des langages de programmation[67].

Un dernier sillon de cette recherche rajouta un point intéressant. Il toucha une dimension importante de la nature des fonctions, à savoir leurs arguments, c'est-à-dire les symboles qui représentent ce sur quoi elles sont appliquées. Dans le langage informatique classique, ces arguments des fonctions sont nommés des intrants ou des données (*data*). Or, des fonctions peuvent être définies de façon extensionnelle, c'est-à-dire, soit par la présentation descriptive des éléments qui constituent une relation fonctionnelle particulière ou soit plus simplement sous la forme de liste. Dans un cadre informatique, si la présentation extensionnelle est complexe, on imposera une structure aux données (*data structure*). C'est souvent sous cette forme qu'elle sera concrètement implémentée dans une base de données : « La programmation porte essentiellement sur certaines "structures de données" et les fonctions entre elles[68]. »

Ainsi, cette dernière voie de recherche démontra, par un cheminement inverse, des équivalences entre des fonctions calculables du lambda-calcul, des algorithmes, des automates, des grammaires catégorielles, des langages de programmation, des structures de données et certains de type de structures mathématiques (algèbre, topologie, géométrie, treillis, etc.[69]). Autrement dit, ce que l'un peut calculer, l'autre peut aussi le faire. On trouve cette idée chez Rodney Burstall et Peter J. Landin :

> Une programmation porte essentiellement sur certaines « structures de données » [*data structures*] et sur des fonctions entre ces structures. [...] Ayant commencé avec des structures familières telles les groupes et les anneaux, les algébristes ont développé une notion plus approfondie des structures algébriques (ou algèbre) qui inclut celles-ci comme exemples, mais qui inclut aussi plusieurs des entités qui dans un ordinateur sont pensées comme des structures de données[70].

Richard Büchi écrit également :

> Si la définition d'un « automate fini » est correctement choisie, il s'avère que tous les concepts de base et les résultats concernant la structure et le comportement des automates finis sont en fait seulement des cas spéciaux des concepts fondamentaux (homomorphisme, relation de congruence, algèbre libre) et des faits d'algèbre abstraite. La théorie des automates est simplement la théorie de l'algèbre universelle [...] avec des opérations unaires, et une mise en évidence sur les algèbres de type/groupe fini[71].

Il faudra cependant attendre Dana A. Scott pour expliciter clairement les fondements mathématiques de la nature des fonctions computables. Dans un article de 1970, il soulignera la faiblesse des théories générales des fonctions : « La principale nouveauté mathématique de la présente étude est la création d'une théorie mathématique des fonctions adéquates qui accomplit ces objectifs (systématiquement!)[72] [...] »

Malgré la richesse de ces liens, de nombreux problèmes persistent. Autant la théorie des algèbres abstraites que la théorie classique de la computation ne semblent pas les aborder de manière satisfaisante. On pense ici à des problèmes d'approximation, d'optimisation, de données massives, de la complexité, de la dynamicité et surtout de non-compositionnalité de plusieurs fonctions. Il faudra attendre les développements récents des mathématiques pour que ces dimensions particulières soient reliées à des structures mathématiques.

Ainsi, pour qu'un modèle formel soit possiblement mis en relation avec un modèle computationnel, il faut que le langage formel utilisé exprime des structures mathématiques fonctionnelles. Mais encore faut-il que ces structures permettent la calculabilité. Cependant, si elles possèdent cette propriété, alors comme l'a montré le précédent résumé de la recherche en informatique théorique, il existe une équivalence entre certaines structures mathématiques calculables et de multiples langages exprimant des fonctions. Sachant cette équivalence, il sera alors possible pour la recherche scientifique de traduire leurs modélisations formelles directement en une modélisation computationnelle. Aussi verra-t-on souvent des modèles algébriques de grandes complexités exprimer directement les fonctions qu'ils contiennent en algorithmes ou en un langage de programmation adéquat. Et la raison en serait que l'on connaît maintenant mieux les fondements mathématiques de ces approches computationnelles. Comme l'écrit Mahoney :

> En l'absence de structures mathématiques qui autorisent l'abstraction et la généralisation, les modèles computationnels ne disent pas grand-chose. Pas plus qu'ils ne fonctionnent comme les modèles traditionnels qui ont fourni une compréhension de la nature sur la base de ce que nous pouvons tester nos connaissances en faisant avancer les choses dans le monde[73].

En somme, on voit mieux comment un modèle formel est contraint en aval par la computation. Pour que les équations ou formules qu'utilise un modèle formel soient pertinentes pour une modélisation computationnelle, il faut que les relations structurantes possèdent des relations de dépendances fonctionnelles. Cette dépendance fonctionnelle est une condition *sine qua non*, de la traduction computationnelle, c'est-à-dire en algorithme ou autres « procédures effectives ».

Comme nous le verrons dans la section suivante, ces recherches auront une portée capitale pour une sémiotique computationnelle.

6. Critiques et limites de la modélisation formelle dans une sémiotique computationnelle

La sémiotique computationnelle fera nécessairement l'objet de différentes critiques. Par la mobilisation de l'informatique dans sa démarche, elle se positionne dans une épistémè positiviste, empirique et possiblement expérimentale. Cela n'ira pas sans soulever les critiques classiques élaborées par rapport à ce type d'épistémè. Entre autres critiques, on pourrait voir la sémiotique computationnelle comme (a) substituant le quantitatif au qualitatif, (b) éliminant la recherche analytique et conceptuelle et (c) réduisant ou simplifiant trop la complexité intrinsèque des artefacts sémiotiques. Je présenterai ici ces possibles critiques en mes termes, tout en visant les points spécifiquement reliés aux modélisations formelle et computationnelle.

La première critique, générale, pouvant être formulée soutiendrait qu'une sémiotique computationnelle est simplificatrice et réductrice parce qu'elle substitue une approche quantitative à une approche qualitative et conceptuelle. Or la réalité de la pratique sémiotique n'est pas si simple. Une telle critique, en effet, relève d'une compréhension discutable de la nature et de la place d'un modèle formel dans une démarche de connaissance scientifique. Telle que formulée, cette critique reprend de manière implicite l'opposition classique de la philosophie des sciences de tradition allemande où l'on oppose les sciences de la nature et les sciences de l'esprit (*Geistwissenschaften*) ou de la tradition anglo-saxonne où on oppose l'empirisme/le positiviste et l'herméneutique. Ces deux oppositions sont vulgarisées en termes de sciences dites quantitatives *versus* qualitatives ou plus facilement encore de sciences dures et de sciences molles.

Mais comme le montre la philosophie contemporaine, cette vision ne correspond pas à la pratique concrète des sciences de quelque nature qu'elles soient. En effet les sciences sont des pratiques épistémiques qui ne peuvent être comprises comme opérant dans une seule et unique de ces positions épistémologiques. Au contraire une science ne peut approcher son objet d'étude que par le biais d'une multitude de points de vue ou, dit plus techniquement, de modèles interreliés. Aucune science dans sa démarche ne repose sur un modèle unique. Comme le formule Sara Green : « un seul modèle est insuffisant[74]. »

Il en va de même en sémiotique : elle ne peut pas reposer que sur un seul et unique modèle. Une étude un peu attentive de ses pratiques montre rapidement que la sémiotique met en place de multiples modèles dont certains peuvent être vus comme modélisant du « qualitatif » et d'autres du « quantitatif », et où chacune des approches ne s'y investit pas avec la même énergie épistémique.

De fait, en raison de la complexité de ses objets d'étude, la sémiotique se concentrera classiquement sur un type de modèle plutôt qu'en autre. On conviendra facilement que celui qu'elle privilégie est le modèle « qualitatif » de type conceptuel. Celui-ci met en action des procédures épistémiques de type analytique et synthétique, des descriptions et des explications qui en appelleront à de nombreux paradigmes théoriques ; le but de ce modèle étant de construire une première conceptualisation des multiples relations qui structurent un artefact sémiotique et des interprétations qui peuvent lui être associées pour finalement les exprimer dans un langage naturel. Mais, on ne s'arrêtera pas là. Très souvent, pour confirmer la validité des énoncés produits dans ce modèle, on en appellera à diverses méthodes de validation. Par exemple, par déduction on validera les hypothèses analytiques et interprétatives en mettant en correspondance des cas, des situations, concrètes, etc. Ceux-ci ne sont pas que des « illustrations », et ils serviront d'échantillons de validation empirique. Ou encore, l'hypothèse sera considérée vraie jusqu'à la rencontre d'un contre-exemple ou jusqu'à preuve du contraire. Dans certaines autres recherches, on procédera par induction et abduction. On partira des exemples ou des cas et on tentera par généralisation de trouver des « régularités » tant dans les structures des artefacts que dans leurs interprétations.

Or, il faut bien comprendre que ces méthodes de validation, en apparence simples et uniquement qualitatives, cachent habilement du quantitatif. En effet, les syntagmes comme « des cas », « un contre-exemple » ou « quelques situations » contiennent des marqueurs linguistiques (des articles déterminatifs) qui expriment linguistiquement de la quantification. Ces marqueurs signifient qu'il y a un certain *nombre* (bien que souvent limité) de situations ou de cas qui valident l'hypothèse. Bien que la méthode de validation ne soit pas exprimée formellement et ne mette pas en jeu des « expérimentations », il a y techniquement appel à du quantitatif, si minimal soit-il.

Par conséquent, il est trop général de soutenir qu'une modélisation computationnelle est réductrice parce qu'elle substitue une approche quantitative à une approche qualitative. Aucune personne en sémiotique ne dira que ses énoncés à propos des artefacts sémiotiques sont vrais uniquement parce que c'est elle qui les a déduits « analytiquement » de prémisses, de postulats ou d'axiomes ou encore que ces énoncés sont vrais par ce que c'est elle qui en a fait la synthèse et les généralisations. Aussi en appellera-t-on plutôt à des validations externes qui souvent seront « des » exemples ou « des » contre-exemples. Malheureusement ce type de formulation masque l'utilisation du quantitatif à travers le flou du langage naturel.

Par ailleurs, et ceci est plus important, il faut se rappeler que dans une perspective bourbakienne, la sémantique d'un modèle formel de type mathématique renvoie avant tout à des structures dans l'univers de référence. Or ces structures dites mathématiques sont avant tout des relations de divers types entre des éléments qui dans certains cas peuvent être certes, des nombres, mais qui dans plusieurs autres cas peuvent être tout autre chose. Une belle illustration de cette formalisation autre que « quantitative » est la modélisation formelle de type grammaire catégorielle que Lambek et Lambek ont appliqué à des relations de parenté[75]. Une grammaire générative (à la Chomsky) ou une grammaire fonctionnelle (à la Halliday) est une hypothèse de type mathématique non quantitative sur des relations structurelles dans la langue. Et on pourrait montrer que de nombreuses analyses de types géométriques utilisent des chiffres comme des artifices intermédiaires pour assister des calculs sur des vecteurs qui représentent des relations dans des phénomènes sémiotiques complexes. L'exemple donné plus haut sur la relation

Fils de montre qu'une structure mathématique de relation de dépendance fonctionnelle peut être projetée sur des personnes existant dans le monde réel.

La critique qui soutient qu'une sémiotique computationnelle qui en appelle à des modèles formels de type mathématique privilégie une approche quantitative aux dépens d'une approche qualitative ne correspond pas parfaitement aux pratiques effectives en sémiotique ou encore à la nature exacte d'un modèle formel. D'une part, la sémiotique utilise souvent de manière discrète sinon même implicite de la validation à teneur quantitative même si cela est minimal. D'autre part, une telle critique ne semble pas avoir compris que la modélisation formelle de type mathématique peut porter sur autre chose que des nombres tel que par exemple des structures de relation entre des éléments de toute sorte.

Une deuxième critique pourrait être élaborée. Plus complexe et problématique, elle soutiendrait que la sémiotique computationnelle élimine la modélisation analytique et conceptuelle. Cette critique me semble problématique, car elle ne correspond pas à une contrainte inhérente à la construction même de modèle formel. En effet, comme je l'ai montré plus haut, dans une démarche scientifique un modèle formel est contraint par ses relations aux autres modèles. Il ne se construit pas de manière indépendante et isolée des autres modèles qui participent à la connaissance. Par exemple, un modèle formel dépend en amont et en aval d'un modèle conceptuel sur lequel il est appliqué : dit autrement, un modèle formel ne peut construire des énoncés formels ni les interpréter qu'en relation à ce qui est conceptualisé à propos de l'objet d'étude. Un modèle formel est donc intimement contraint par un modèle conceptuel. Sans cette sémantique conceptuelle, un modèle formel est sans signification et, de ce fait, incompréhensible.

Il s'agit d'une illusion créée par un habile marketing de la science ou de vulgarisations simplificatrices que de penser qu'on puisse prendre des formules mathématiques clé en main et les appliquer à n'importe quel objet. La loi de l'attraction des masses, quelle qu'en soit la formalisation, ne peut expliquer l'attraction d'une personne pour une autre! L'équation $F = Ma$ a des référents très spécifiques qui, dans le modèle conceptuel, sont nommés *masse et accélération et force*. Ce sont là des concepts qui reposent sur une conceptualisation technique et complexe et qui n'est pas simple et facile à comprendre. Ces concepts n'ont rien à faire avec les concepts qui décrivent une attraction amoureuse entre des personnes.

Compte tenu de la complexité des objets d'étude de son domaine d'étude, une sémiotique dite computationnelle ne peut éliminer le travail analytique et interprétatif qui doit se construire dans un modèle conceptuel. Au contraire, il met une pression sur ce modèle dont la fonction est d'explorer et de préciser au meilleur de ses moyens les concepts et la conceptualisation de l'objet d'étude afin de le rendre accessible à un éventuel modèle formel. C'est en effet sur le contenu du modèle conceptuel que le modèle formel projette des hypothèses structurantes. Même dans les cas où un modèle formel est offert *a priori*, ce sera le modèle conceptuel qui permettra d'interpréter les équations et les formules que propose une hypothèse formelle.

En somme, la modélisation formelle en sémiotique ne peut éliminer la modélisation conceptuelle. Aucune modélisation sémiotique computationnelle n'est possible si le modèle formel qui sous-tend les algorithmes n'est pas relié en amont, en parallèle ou en aval, à un travail épistémique conceptuel et interprétatif.

Une troisième critique plus profonde pourrait être adressée. Elle limiterait absolument une sémiotique computationnelle, mais du même coup en préciserait absolument les cibles. Elle soutiendrait que la modélisation formelle à horizon computationnel ne peut rendre compte de la complexité du sémiotique. De ce point de vue, une sémiotique computationnelle serait donc réductrice.

Un modèle formel qui, en aval, doit être éventuellement intégré dans une technologie informatique dépend effectivement des contraintes que lui impose un modèle computationnel. En effet, un modèle formel à horizon computationnel ne peut se concentrer que sur des relations structurantes de types dépendances fonctionnelles et qui sont calculables. Car c'est le seul type de structures qui peuvent être traduites dans des algorithmes. Autrement dit, une équation ou une formule d'un langage formel est traduisible en algorithmes si et seulement si elle est une fonction qui peut être calculée par des procédures effectives computables. Dit banalement, une fonction non calculable ne peut être computable! En conséquence, un modèle formel sémiotique si bien formé soit-il, si élégant soit-il, est inutile si la grammaire ou la manipulation des symboles exprimant des structures fonctionnelles ne permet pas des procédures effectives, c'est-à-dire du calcul ou de la computation. Ceci limite radicalement la construction de modèle formel pour une sémiotique computationnelle. Qui plus est, rien ne garantit que dans un artefact sémiotique seules les structures de relations de dépendances fonctionnelles soient les seules pertinentes ou intéressantes.

En d'autres termes, un modèle formel couplé à un modèle computationnel est limité. Il impose une vision spécifique de l'objet d'étude. Par sa nature même, un tel modèle est une sorte d'idéalisation[76]. Il impose des choix. En conséquence, il est toujours réducteur. Il n'a jamais le type de richesse épistémique que possède un modèle conceptuel. Par exemple, un modèle formel mathématique de type *logique* qui contient des symboles atomiques et des règles spécifiques d'inférence exigera une atomisation et une discrétisation des éléments. Il ne se focalisera que sur des structures statiques consolidées. Il lui sera difficile, de par son formalisme même[77], de traiter le continu et la dynamique de certains phénomènes[78].

Un modèle formel mathématique sera encore des plus réducteurs. En effet comme celui-ci ne peut cibler que des structures relationnelles de dépendances fonctionnelles computables, il risque de rencontrer le mur de la non-computationnalité. Car, comme on le sait par Godel, tout système formel présente la non-complétude[79]. Et on sait par Turing qu'il n'est pas possible de déterminer quand un algorithme s'arrêtera. Par Chaitin, on sait que plus un système est complexe, plus il tend vers le non-computable[80]. En conséquence, un modèle formel si bien formé soit-il, peut s'avérer non traduisible en algorithmes parce que non computable. Voici un exemple presque contre-intuitif : une équation simple, élégante et ayant la forme suivante : $ax + bx = c$ (dite équation diophantienne) a été démontrée non computable[81]! Et il existe le nombre infini de de telles fonctions non calculables. Une sémiotique computationnelle est donc absolument limitée. Elle ne peut pas traiter cette infinité de fonctions! Comme son nom l'indique : elle ne traite que des fonctions « computationnelles ».

Il s'avère cependant qu'il reste encore une infinité de fonctions calculables. Assez pour occuper aussi une infinité de recherches en sémiotique. La critique qui soutiendrait que la sémiotique computationnelle est limitée et réductrice est incontournable, mais on peut la voir positivement : elle définit précisément la cible. Une sémiotique computationnelle, au sens strict, ne peut avoir comme objets d'étude que des artefacts sémiotiques présentant des structures relationnelles formalisables et computables. Et comme il est pensable que plusieurs des relations qui tissent les artefacts sémiotiques ne puissent être formalisables et computables, il s'ensuit qu'une sémiotique computationnelle ne pourra jamais être totalement automatique et algorithmique.

En conséquence l'informatique devra accepter de n'être qu'un assistant dans la recherche sémiotique. La sémiotique computationnelle devra faire preuve d'humilité et accepter de ne travailler que sur ce qui est formalisable et computable. Il n'en reste pas moins que ce domaine de recherche demeure vaste. Les deux disciplines pourront naviguer dans deux infinis : l'infini du computable et l'infini du non-computable. Là est l'un des paradoxes de leur rencontre.

Bibliographie

ALBUS, James S. & Alexander M. MEYSTEL, « A Reference Model Architecture for Design and Implementation of Intelligent Control in Large and Complex Systems », *International Journal of Intelligent Control and Systems*, vol. 1, no 1, 1996, p. 15-30.

ARMATTE, Michel & Amy DAHAN DALMEDICO, « Modèles et modélisations, 1950-2000 : nouvelles pratiques, nouveaux enjeux », *Revue d'histoire des sciences*, tome 57, no 2, 2004, p. 243-303.

ASPERTI, Andrea & Giuseppe LONGO, *Categories, Types and Structures*, Cambridge, The MIT Press, 1991.

AUDIN, Michèle, « Hommage à Claude Lévi-Strauss », *Images des mathématiques*, 2009. En ligne : <https://images.math.cnrs.fr/Hommage-a-Claude-Levi-Strauss.html>.

BARBUT, Marc, « Sur le sens du mot "structure" en mathématiques », *Les Temps Modernes*, no 246, 1966, p. 791-814.

BARTHES, Roland, *Système de la mode*, Paris, Seuil, 1967.

BERNARD, Michel & Baptiste BOHET, *Littérométrie : outils numériques pour l'analyse des textes littéraires*, Paris, Presses Sorbonne nouvelle, 2017.

BOUDON, Raymond, *À quoi sert la notion de « structure »? Essai sur la signification de la notion de structure dans les sciences humaines*, Paris, Gallimard, 1968.

BOURBAKI, Nicolas, *Théorie des ensembles*, Berlin, Springer, 2006 [1970].

BRANDT, Line, *The Communicative Mind: A Linguistic Exploration of Conceptual Integration and Meaning Construction*, Newcastle, Cambridge Scholars Publishing, 2013.

BRANDT, Per Åge, « Toward a Cognitive Semiotics », *Recherches en communication*, no 19, 2003, p. 21-34.

BRIER, Søren, « Biosemiotics and the Foundation of Cybersemiotics: Reconceptualizing the Insights of Ethology, Second-Order Cybernetics, and Peirce's Semiotics in Biosemiotics to Create a Non-Cartesian Information Science », *Semiotica*, vol. 127, no 1-4, 1999, p. 169-198.

BRUNET, Étienne, *Méthodes quantitatives et informatiques dans l'étude des textes*, Paris, Champion, 1986.

BÜCHI, J. Richard, « Algebraic Theory of Feedback in Discrete Systems, Part I (1966) », dans *The Collected Works of J. Richard Büchi*, textes rassemblés par S. Mac Lane & D. Siefkes, New York, Springer, 1990, p. 338-369.

BUNDGAARD, Peer & Frederik STJERNFELT, *Signs and Meaning: Five Questions*, New York, Automatic Press, 2009.

BURDICK, Anne, Johanna DRUCKER, Peter LUNENFELD, Todd PRESNER & Jeffrey SCHNAPP, *Digital Humanities*, Cambridge, The MIT Press, 2012.

BURSTALL, Rodney M. & Peter J. LANDIN, « Programs and Their Proofs: An Algebraic Approach », *Machine Intelligence*, vol. 4, 1969, p. 17-44.

CHAITIN, Gregory, Francisco A. DORIA & Newton DA COSTA, *Gödel's Way: Exploits into an Undecidable World*, Boca Raton, CRC Press, 2012.

CHARTIER, Jean-François, Davide PULIZZOTTO, Louis CHARTRAND & Jean-Guy MEUNIER, « A Data-Driven Computational Semiotics: The Semantic Vector Space of Magritte's Artworks », *Semiotica*, no 230, 2019, p. 19-69.

CHOMSKY, Noam, *Syntactic Structures*, La Haye, Mouton, 1957.

—, « On certain formal properties of grammars », *Information and Control*, vol. 2, no 2, 1959, p. 137-167.

—, « Trois modèles de description du langage », *Langages*, no 9, 1968, p. 51-76.

CHURCH, Alonzo, « An Unsolvable Problem of Elementary Number Theory », *American Journal of Mathematics*, vol. 58, no 2, 1936, p. 345-363.

COMPAGNO, Dario (dir.), *Quantitative Semiotic Analysis*, New York, Springer, 2018.

CURRY, Haskell B. & Robert FEYS, *Combinatory Logic. Volume 1*, Amsterdam, North-Holland Publishing Company, 1958.

DAVIS, Martin, *Computability and Unsolvability*, New York, Dover Publications, 1982.

DE FRANCESCHI, Élisabeth, « Le moment de conclure (1977-1978) », *Oxymoron, revue des médiations thérapeutiques par l'art*, no 0, 2010. En ligne : <http://revel.unice.fr/oxymoron/index.html?id=3072>.

DE SOUZA, Clarisse Sieckenius, *The Semiotic Engineering of Human-Computer Interaction*, Cambridge, The MIT Press, 2005.

DESCLÉS, Jean-Pierre, « Réseaux sémantiques : la nature logique et linguistique des relateurs », *Langages*, no 87, 1987, p. 55-78.

—, « La Grammaire Applicative et Cognitive construit-elle des représentations universelles? », *Linx*, no 48, 2003, p. 139-160.

DIMINESCU, Dana & Michel WIEVIORKA, « Le défi numérique pour les sciences sociales », *Socio*, no 4, 2015, p. 9-17.

DUHEM, Pierre, *La théorie physique. Son objet et sa structure*, Paris, Chevalier & Rivière, 1906.

FETZER, James H., « Minds and Machines: Limits to Simulations of Thought and Action », *International Journal of Signs and Semiotic Systems*, vol. 1, no 1, 2011, p. 39-48.

GANDY, Robin, « Church's Thesis and Principles for Mechanisms », *Studies in Logic and the Foundations of Mathematics*, vol. 101, « The Kleene Symposium », 1980, p. 123-148.

GIERE, Ronald, « Using Models to Represent Reality », dans L. Magnani, N. J. Nersessian & P. Thagard (dir.), *Model-Based Reasoning in Scientific Discovery*, Boston, Springer, 1999, p. 41-57.

GÖDEL, Kurt, *On Formally Undecidable Propositions of Principia Mathematica and Related Systems*, trad. de l'allemand par B. Meltzer, New York, Dover Publications, 1992 [1931].

GRANGER, Gilles-Gaston, *Pensée formelle et sciences de l'homme*, Paris, Aubier-Montaigne, 1967.

GREEN, Sara, « When one Model is not Enough: Combining Epistemic Tools in Systems Biology », *Studies in History and Philosophy of Biological and Biomedical Sciences*, vol. 44, no 2, 2013, p. 170-180.

GREIMAS, Algirdas J. & Joseph COURTÉS, *Sémiotique : dictionnaire raisonné de la théorie du langage*, Paris, Hachette, 1979.

GROSS, Maurice & André LENTIN, *Notions sur les grammaires formelles*, Paris, Gauthier-Villars, 1970.

GROUPE µ, *Principia Semiotica : aux sources du sens*, Bruxelles, Les Impressions nouvelles, 2015.

GUDWIN, Ricardo & Fernando GOMIDE, « Computational Semiotics: An Approach for the Study of Intelligent Systems. Part I: Foundations », rapport, 1997. En ligne : <http://www.dca.fee.unicamp.br/~gudwin/ftp/publications/rep1_97.pdf>.

HILBERT, David, *Les fondements de la géométrie*, trad. de l'allemand par P. Rossier, Paris, Dunod, 1971 [1899].

HINTIKKA, Jaakko, « On the Development of the Model-Theoretic Viewpoint in Logical Theory », *Lingua Universalis vs. Calculus Ratiocinator An Ultimate Presupposition of Twentieth-Century Philosophy*, Dordrecht, Springer, 1997, p. 104-161.

HJEMSLEV, Louis, *Prolégomènes à une théorie du langage, suivi de la structure fondamentale du langage*, trad. du danois par U. Canger, avec la collab. d'A. Wewer, Paris, Minuit, coll. « Arguments », 1968.

HOFFMEYER, Jesper, « Surfaces Inside Surfaces. On the Origin of Agency and Life », *Cybernetics & Human Knowing*, vol. 5, no 1, 1998, p. 33-42.

ISRAEL, Giorgio, *La Mathématisation du réel. Essai sur la modélisation mathématique*, Paris, Seuil, 1996.

KETNER, Kenneth Laine, « Peirce and Turing: Comparisons and Conjectures », *Semiotica*, vol. 68, no 1-2, 1988, p. 33-62.

KULL, Kalevi & Ekaterina VELMEZOVA, « What is the Main Challenge for Contemporary Semiotics? », *Sign System Studies*, vol. 42, no 4, 2014, p. 530-548.

LACOMBE, Daniel, « La théorie des fonctions récursives et ses applications », *Bulletin de la Société Mathématique de France*, tome 88, 1960, p. 393-468.

LADRIÈRE, Jean, *Les limitations internes des formalismes*, Paris, Gauthier-Villars, 1957.

LAMBEK, Joachim & Michael LAMBEK, « The Kinship Terminology of Malagasy Speakers in Mayotte », *Anthropological Linguistics*, vol. 23, no 4, 1981, p. 154-182.

LANDIN, Peter J., « The Next 700 Programming Languages », *Communications of the ACM*, vol. 9, no 3, 1966, p. 157-166.

LÉVI-STRAUSS, Claude, « Réflexions sur l'atome de parenté », *L'Homme. Revue française d'anthropologie*, tome 13, no 3, 1973, p. 5-30.

LONGHI, Julien, « Contours, perspectives et tensions des "humanités numériques" », *Sens-Dessous*, no 24, 2019, p. 123-135.

LUHMANN, Niklas, *Social Systems*, trad. de l'allemand par J. Bednarz Jr. & D. Barcker, préface d'E. M. Knodt, Stanford, Stanford University Press, 1995.

LUKASIEWICZ, Jan, *Elements of Mathematical Logic*, New York, MacMillan, 1963 [1929].

MAHONEY, Michael S., « The Structures of Computation and the Mathematical Structure of Nature », *The Rutherford Journal*, vol. 3, 2010. En ligne : <http://www.rutherfordjournal.org/article030107.html>.

MANNING, Peter K. « Semiotics and Data Analysis », dans M. Hardy & A. Bryman (dir.), *Handbook of Data Analysis*, Londres, SAGE Publications, 2004, p. 566-587.

MARKOV, Andreï, « The Theory of Algorithms », *American Mathematical Society Translations: Series 2*, vol. 15, 1960, p. 1-14.

MARR, David, « Selections from Vision », dans A. Noë & E. Thompson (dir.), *Vision and Mind. Selected Readings in the Philosophy of Perception*, Cambridge, The MIT Press, 2002, p. 229-265.

MARTY, Robert, *L'Algèbre des signes. Essai de sémiotique scientifique d'après C.S. Peirce*, Amsterdam/Philadelphie, John Benjamins Publishing Company, 1990.

MAYAFFRE, Damon, *Le discours présidentiel sous la V^e^ République. Chirac, Mitterrand, Giscard, Pompidou, de Gaulle*, Paris, Presses de Sciences Po, 2012.

McCARTHY, John, « Recursive Functions of Symbolic Expressions and their Computation by Machine – Part 1 », *Communications of the ACM*, vol. 3, no 4, 1960, p. 184-195.

—, « A Basis for a Mathematical Theory of Computation », *Proceedings of the Western Joint Computer Conference*, 1961, p. 225-238.

MEUNIER, Jean-Guy, « Artificial Intelligence and Sign Theory », *Semiotica*, vol. 77, no 1-3, 1989, p. 43-64.

—, « Categorial Structure of Iconic Languages », *Theory and Psychology*, vol. 8, no 6, 1998, p. 805-827.

—, « Vers une sémiotique computationnelle? », *Applied Semiotics/Sémiotique appliquée*, no 26, 2018, p. 75-107. En ligne : <http://french.chass.utoronto.ca/as-sa/ASSA-No26/26-6.pdf>.

MEUNIER, Jean-Guy, Ismail BISKRI & Dominic FOREST, « Classification and Categorization in Computer Assisted Reading and Analysis of Texts », dans C. Lefebvre & H. Cohen (dir.), *Handbook of Categorization in Cognitive Science*, New York, Elsevier, 2005, p. 955-978.

MONTAGUE, Richard, *Formal Philosophy: Selected Papers of Richard Montague*, New Haven, Yale University Press, 1974.

MORGAN, Mary S. & Margaret MORRISON (dir.), *Models as Mediators. Perspectives on Natural and Social Science*, Cambridge, Cambridge University Press, 1999.

MORRIS, Charles W., *Foundations of the Theory of Signs*, Chicago, The University of Chicago Press, 1938.

NADIN, Mihai, « Information and Semiotic Processes. The Semiotics of Computation », *Cybernetics and Human Knowing*, vol. 18, no 1-2, 2011, p. 153-175.

—, « Reassessing the Foundations of Semiotics: Preliminaries », *International Journal of Signs and Semiotic Systems*, vol. 2, no 1, 2012, p. 1-31.

PERLOVSKY, Leonid I., *Neural Networks and Intellect: Using Model-Based Concepts*, New York, Oxford University Press, 2001.

PETITOT, Jean, *Morphogénèse du sens*, Paris, Presses universitaires de France, 1985.

PIOTROWSKI, David & Yves-Marie VISETTI, « Connaissance sémiotique et mathématisation. Sémiogenèse et explicitation », *Versus*, no 118, 2014, p. 141-170.

POST, Emil L., « Recursively Enumerable Sets of Positive Integers and their Decision Problems », *Bulletin of the American Mathematical Society*, vol. 50, no 5, 1944, p. 284-316. En ligne : <https://projecteuclid.org/download/pdf_1/euclid.bams/1183505800>.

PRIETO, Luis, *Messages et signaux*, Paris, Presses universitaires de France, 1966.

PULIZZOTTO, Davide, *La sémiotique computationnelle : entre narrativité et apprentissage automatique. Une démonstration de faisabilité avec un corpus automatique journalistique à propos du Printemps érable*, thèse de doctorat, Montréal, Université du Québec à Montréal, 2020.

RAPAPORT, William J., « Semiotic Systems, Computers, and the Mind: How Cognition Could Be Computing », *International Journal of Signs and Semiotic Systems*, vol. 2, no 1, 2012, p. 32-71.

RASTIER, François, *La mesure et le grain : sémantique de corpus*, Paris, Honoré Champion, 2011.

RASTIER, François, Marc CAVAZZA & Anne ABEILLÉ, *Sémantique pour l'analyse de la linguistique à l'informatique*, Paris, Masson, 1994.

RIEGER, Burghard B., « Computing Fuzzy Semantic Granules from Natural Language Texts. A computational semiotics approach to understanding word meanings », *Proceedings of the IASTED International Conference Artificial Intelligence and Soft Computing*, 1999, p. 475-479. En ligne : <https://www.uni-trier.de/fileadmin/fb2/LDV/Rieger/Publikationen/Aufsaetze/99/iasted99.pdf>.

ROBERTS, Don D., *The Existential Graphs of Charles S. Peirce*, La Haye, Mouton, 1973.

SCHÜTZENBERGER, Marcel-Paul, « Some Remarks on Chomsky's Context-Free Languages », *Quarterly Progress Report of the MIT Research Laboratory of Electronics*, no 68, 1961, p. 155-170.

SCOTT, Dana, « Outline of a Mathematical Theory of Computation », *Technical Monograph PRG-2*, Oxford University Computing Laboratory, 1970, p. 169-176.

SEBEOK, Thomas A., *Biosemiotics*, Berlin, Mouton de Gruyter, 1992.

SHANIN, Teodor, « Models and Thought », dans T. Shanin (dir.), *The Rules of the Game*, Hoboken, Taylor and Francis, 2013 [1972], p. 1-22.

SHAUMYAN, Sebastian K., *Applicational Grammar as a Semantic Theory of Natural Languages*, Chicago, Chicago University Press, 1977.

STRACHEY, Christopher, « Towards a Formal Semantics », dans T. B. Steel (dir.), *Formal Language Description Languages for Computer Programming*, Amsterdam, North-Holland Publishing Company, 1966, p. 198-220.

STRAWSON, Peter Frederick, « On referring », *Mind*, vol. 59, no 235, 1950, p. 320-344.

SUÁREZ, Mauricio & Albert SOLÉ, « On the Analogy between Cognitive Representation and Truth », *Theoria*, vol. 21, no 1, 2006, p. 39-48. En ligne : <https://www.ehu.eus/ojs/index.php/THEORIA/article/view/552/401>.

TANAKA-ISHII, Kumiko, *Semiotics of programming*, New York, Cambridge University Press, 2010.

TIJUS, Charles, Javier BARCENILLA, Javier CAMBON DE LAVALETTE & Jean-Guy MEUNIER, « The Design, Understanding and Usability of Pictograms », dans D. Alamargot, P. Terrier & J.-M. Cellier (dir.), *Improving the Production and*

Understanding of Written Documents in the Workplace, Amsterdam, Elsevier, 2007, p. 17-32.

TURING, Alan M., « On Computable Numbers, with an Application to the Entscheidungsproblem », *Proceedings of the London Mathematical Society*, vol. 42, no 1, 1937, p. 230-265.

von BERTALANFFY, Ludwig, « Conclusion », *Human Biology*, vol. 23, no 4, 1951, p. 336-345.

—, « Problems of General System Theory », *Human Biology*, vol. 23, no 4, 1951, p. 302-312.

—, « Towards a Physical Theory of Organic Teleology: Feedback and Dynamics », *Human Biology*, vol. 23, no 4, 1951, p. 346-361.

von NEUMANN, John, « The general and logical theory of automata (1951) », dans *John von Neumann - Collected Works*, vol. 5, textes rassemblés par A. H. Taub, Oxford, Pergamon Press, 1961, p. 288-326.

WILDGEN, Wolfgang, *Catastrophe Theoretic Semantics. An Elaboration and Application of René Thom's Theory*, Amsterdam, John Benjamins Publishing Company, 1982.

ZEMANEK, Heinz, « Semiotics and Programming Languages », *Communications of the ACM*, vol. 9, no 3, 1966, p. 139-143.

Notes

1 C. LÉVI-STRAUSS, « Réflexions sur l'atome de parenté », *L'Homme. Revue française d'anthropologie*, tome 13, no 3, 1973, p. 5.

2 P. BUNDGAARD & F. STJERNFELT, *Signs and Meaning: Five Questions*, New York, Automatic Press, 2009.

3 En 1972, Lotman écrivait, je traduis : « La capacité des différentes disciplines mathématiques de servir de métalangage aussi dans la description de phénomènes artistiques est évidente » (« The ability of various mathematical disciplines to serve as a metalanguage also in the description of the phenomena of art is evident. »), cité dans K. KULL & E. VELMEZOVA, « What is the Main Challenge for Contemporary Semiotics? », *Sign System Studies*, vol. 42, no 4, 2014, p. 531.

4 *Ibid*.

5 Il y avait bien dans le courant structuraliste des réflexions théoriques sur cette notion de structure (voir R. BOUDON, *À quoi sert la notion de « structure »? Essai sur la signification de la notion de structure dans les sciences humaines*, Paris, Gallimard, 1968). Elle n'en demeurait pas moins analytique et conceptuelle. Quelques mathématiciens cependant l'avaient explorée sur le plan formel des sciences appliquées en sciences sociales et humaines : voir M. BARBUT, « Sur le sens du mot "structure" en mathématiques », *Les Temps Modernes*, no 246, 1966, p. 791-814.

6 M. NADIN, « Reassessing the Foundations of Semiotics: Preliminaries », *International Journal of Signs and Semiotic Systems*, vol. 2, no 1, 2012, p. 1-31, cité dans K. KULL & E. VELMEZOVA, « What is the Main Challenge for Contemporary Semiotics? », *loc. cit.*, p. 532.

7 Voir à ce sujet D. D. ROBERTS, *The Existential Graphs of Charles S. Peirce*, La Haye, Mouton, 1973.

8 R. MARTY, *L'Algèbre des signes. Essai de sémiotique scientifique d'après C.S. Peirce*, Amsterdam/Philadelphie, John Benjamins Publishing Company, 1990.

9 L. HJEMSLEV, *Prolégomènes à une théorie du langage, suivi de la structure fondamentale du langage*, trad. du danois par U. Canger, avec la collab. d'A. Wewer, Paris, Minuit, coll. « Arguments », 1968.

10 C. W. MORRIS, *Foundations of the Theory of Signs*, Chicago, The University of Chicago Press, 1938.

11 Voir notamment S. K. SHAUMYAN, *Applicational Grammar as a Semantic Theory of Natural Languages*, Chicago, Chicago University Press, 1977 ; J.-P. DESCLÉS, « La Grammaire Applicative et Cognitive construit-elle des représentations universelles? », *Linx*, no 48, 2003, p. 139-160.

12 M. AUDIN, « Hommage à Claude Lévi-Strauss », *Images des mathématiques*, 2009. En ligne : <https://images.math.cnrs.fr/Hommage-a-Claude-Levi-Strauss.html>.

13 J. LAMBEK & M. LAMBEK, « The Kinship Terminology of Malagasy Speakers in Mayotte », *Anthropological Linguistics*, vol. 23, no 4, 1981, p. 154-182.

14 J.-G. MEUNIER, « Categorial structure of Iconic Languages », *Theory and Psychology*, vol. 8, no 6, 1998, p. 805-827 ; C. TIJUS *et al.*, « The Design, Understanding and Usability of Pictograms », dans D. Alamargot, P. Terrier & J.-M. Cellier (dir.), *Improving the Production and Understanding of Written Documents in the Workplace*, Amsterdam, Elsevier, 2007, p. 17-32.

15 A. J. GREIMAS & J. COURTÉS, *Sémiotique : dictionnaire raisonné de la théorie du langage*, Paris, Hachette, 1979.

16 É. DE FRANCESCHI, « Le moment de conclure (1977-1978) », *Oxymoron, revue des médiations thérapeutiques par l'art*, no 0, 2010. En ligne : <http://revel.unice.fr/oxymoron/index.html?id=3072> ; R. BARTHES, *Système de la mode*, Paris, Seuil, 1967.

17 J. HINTIKKA, « On the Development of the Model-Theoretic Viewpoint in Logical Theory », *Lingua Universalis vs. Calculus Ratiocinator. An Ultimate Presupposition of Twentieth-Century Philosophy*, Dordrecht, Springer, 1997, p. 104-161.

18 W. WILDGEN, *Catastrophe Theoretic Semantics. An Elaboration and Application of René Thom's Theory*, Amsterdam, John Benjamins Publishing Company, 1982 ; J. PETITOT, *Morphogénèse du sens*, Paris, Presses universitaires de France, 1985 ; L. BRANDT, *The Communicative Mind: A Linguistic Exploration of Conceptual Integration and Meaning Construction*, Newcastle, Cambridge Scholars Publishing, 2013.

19 D. PIOTROWSKI & Y.-M. VISETTI, « Connaissance sémiotique et mathématisation. Sémiogenèse et explicitation », *Versus*, no 118, 2014, p. 141-170.

20 T. A. SEBEOK, *Biosemiotics*, Berlin, Mouton de Gruyter, 1992 ; J. HOFFMEYER, « Surfaces Inside Surfaces. On the Origin of Agency and Life », *Cybernetics & Human Knowing*, vol. 5, no 1, 1998, p. 33-42.

21 S. BRIER, « Biosemiotics and the Foundation of Cybersemiotics: Reconceptualizing the Insights of Ethology, Second-Order Cybernetics, and Peirce's Semiotics in Biosemiotics to Create a Non-Cartesian Information Science », *Semiotica*, vol. 127, no 1-4, 1999, p. 169-198 ; N. LUHMANN, *Social Systems*, trad. de l'allemand par J. Bednarz Jr. & D. Barcker, preface d'E. M. Knodt, Stanford, Stanford University Press, 1995.

22 L. PRIETO, *Messages et signaux*, Paris, Presses universitaires de France, 1966 ; M. NADIN, « Information and Semiotic Processes. The Semiotics of Computation », *Cybernetics and Human Knowing*, vol. 18, no 1-2, 2011, p. 153-175.

23 L. BRANDT, *The Communicative Mind: A Linguistic Exploration of Conceptual Integration and Meaning Construction*, *op. cit.*

24 P. A. BRANDT, « Toward a Cognitive Semiotics », *Recherches en communication*, no 19, 2003, p. 21-34.

25 Cf. notamment H. ZEMANEK, « Semiotics and Programming Languages », *Communications of the ACM*, vol. 9, no 3, 1966, p. 139-143 ; K. L. KETNER, « Peirce and Turing: Comparisons and Conjectures », *Semiotica*, vol. 68, no 1-2, 1988, p. 33-62 ; J.-G. MEUNIER, « Artificial Intelligence and Sign Theory », *Semiotica*, vol. 77, no 1-3, 1989, p. 43-64 ; J. S. ALBUS & A. M. MEYSTEL, « A Reference Model Architecture for Design and Implementation of Intelligent Control in Large and Complex Systems », *International Journal of Intelligent Control and Systems*, vol. 1, no 1, 1996, p. 15-30 ; C. S. DE SOUZA, *The Semiotic Engineering of Human-Computer Interaction*, Cambridge, The MIT Press, 2005 ; K. TANAKA-ISHII, *Semiotics of programming*, New York, Cambridge University Press, 2010 ; J. H. FETZER, « Minds and Machines: Limits to Simulations of Thought and Action », *International Journal of Signs and Semiotic Systems*, vol. 1, no 1, 2011, p. 39-48 ; W. J. RAPAPORT, « Semiotic Systems, Computers, and the Mind: How Cognition Could Be Computing », *International Journal of Signs and Semiotic Systems*, vol. 2, no 1, 2012, p. 32-71.

26 Voir par ex. J. S. ALBUS & A. M. MEYSTEL, « A Reference Model Architecture for Design and Implementation of Intelligent Control in Large and Complex Systems », *loc. cit.* ; R. GUDWIN & F. GOMIDE, « Computational Semiotics: An Approach for the Study of Intelligent Systems. Part I: Foundations », rapport, 1997. En ligne : <http://www.dca.fee.unicamp.br/~gudwin/ftp/publications/rep1_97.pdf> ; B. B. RIEGER, « Computing Fuzzy Semantic Granules from Natural Language Texts. A computational semiotics approach to understanding word meanings », *Proceedings of the IASTED International Conference Artificial Intelligence and Soft Computing*, 1999, p. 475-479. En ligne : <https://www.uni-trier.de/fileadmin/fb2/LDV/Rieger/Publikationen/Aufsaetze/99/iasted99.pdf> ; F. RASTIER, M. CAVAZZA & A. ABEILLÉ, *Sémantique pour l'analyse de la linguistique à l'informatique*, Paris, Masson, 1994 ; L. I. PERLOVSKY, *Neural Networks and Intellect: Using Model-Based Concepts*, New York, Oxford University Press, 2001 ; D. COMPAGNO (dir.), *Quantitative Semiotic Analysis* New York, Springer, 2018.

27 A. BURDICK *et al.*, *Digital Humanities*, Cambridge, The MIT Press, 2012, p. 8.

28 *Ibid.*, p. 16. Je traduis : « Digital Humanities is an extension of traditional knowledge skills and methods, not a replacement for them. »

29 On peut citer par exemple : J. PETITOT, *Morphogénèse du sens*, *op. cit.* ; É. BRUNET, *Méthodes quantitatives et informatiques dans l'étude des textes*, Paris, Champion, 1986 ; J.-P. DESCLÉS, « Réseaux sémantiques : la nature logique et linguistique des relateurs », *Langages*, no 87, 1987, p. 55-78 ; F. RASTIER, M. CAVAZZA & A. ABEILLÉ, *Sémantique pour l'analyse de la linguistique à l'informatique*, *op. cit.* ; B. B. RIEGER, « Computing Fuzzy Semantic Granules from Natural Language Texts », *loc. cit.* ; P. K. MANNING, « Semiotics and Data Analysis », dans M. Hardy & A. Bryman (dir.), *Handbook of Data Analysis*, Londres, SAGE Publications, 2004, p. 566-587 ; J.-G. MEUNIER, I. BISKRI & D. FOREST, « Classification and Categorization in Computer Assisted Reading and Analysis of Texts », dans C. Lefebvre & H. Cohen (dir.), *Handbook of Categorization in Cognitive Science*, New York, Elsevier, 2005, p. 955-978 ; F. RASTIER, *La mesure et le grain : sémantique de corpus*, Paris, Honoré Champion, 2011 ; D. COMPAGNO (dir.), *Quantitative Semiotic Analysis*, *op. cit.* ; J.-G. MEUNIER, « Vers une sémiotique computationnelle? », *Applied Semiotics/Sémiotique appliquée*, no 26, 2018, p. 77. En ligne : <http://french.chass.utoronto.ca/as-sa/ASSA-No26/26-6.pdf> ; J.-F. CHARTIER *et al.*, « A Data-Driven Computational Semiotics: The Semantic Vector Space of Magritte's Artworks », *Semiotica*, no 230, 2019, p. 19-69.

30 J. LONGHI, « Contours, perspectives et tensions des "humanités numériques" », *Sens-Dessous*, no 24, 2019, p. 123-135.

31 M. ARMATTE & A. DAHAN DALMEDICO, « Modèles et modélisations, 1950-2000 : nouvelles pratiques, nouveaux enjeux », *Revue d'histoire des sciences*, tome 57, no 2, 2004, p. 243-303 ; D. DIMINESCU & M. WIEVIORKA, « Le défi numérique pour les sciences sociales », *Socio*, no 4, 2015, p. 9-17.

32 Voir par exemple, pour ne citer que cela, M. S. MORGAN & M. MORRISON (dir.), *Models as Mediators. Perspectives on Natural and Social Science*, Cambridge, Cambridge University Press, 1999 ; R. GIERE, « Using Models to Represent Reality », dans L. Magnani, N. J. Nersessian & P. Thagard (dir.), *Model-Based Reasoning in Scientific Discovery*, Boston, Springer, 1999, p. 41-57.

33 P. F. STRAWSON, « On referring », *Mind*, vol. 59, no 235, 1950, p. 320-344.

34 Von Bertalannfy a si bien formulé la nécessité d'un modèle conceptuel en science : avant d'en arriver à une formalisation mathématique, il faut avoir conceptualisé l'objet même à mathématiser : « Les formulations mathématiques sont des expressions et des conséquences de certains modèles conceptuels que nous appliquons. » Cf. L. VON BERTALANFFY, « Problems of General System Theory », *Human Biology*, vol. 23, no 4, 1951, p. 341 ; L. VON BERTALANFFY, « Conclusion », *Human Biology*, vol. 23, no 4, 1951, p. 336-345 ; L. VON BERTALANFFY, « Towards a Physical Theory of Organic Teleology: Feedback and Dynamics », *Human Biology*, vol. 23, no 4, 1951, p. 346-361.

35 G.-G. GRANGER, *Pensée formelle et sciences de l'homme*, Paris, Aubier-Montaigne, 1967, p. 44 (je souligne).

36 P. DUHEM, *La théorie physique. Son objet et sa structure*, Paris, Chevalier & Rivière, 1906.

37 T. SHANIN, « Models and Thought », dans T. Shanin (dir.), *The Rules of the Game*, Hoboken, Taylor and Francis, 2013 [1972], p. 1. Je traduis : « Scholarly disciplines embody the formalization aspects. [...] It reflects, on the one hand, the need of the human mind for rationalizing and for orientation in an immensely complex reality, to be achieved by building up systematic and coherent general images and symbolic schemes. »

38 J. LADRIÈRE, *Les limitations internes des formalismes*, Paris, Gauthier-Villars, 1957, p. 80. Ladrière synthétise de manière juste, mais pédagogique, les définitions formelles de Peirce, Frege, Russell, Lukscievicks, Husserl, Hilbert, Tarski Gentzen, Bourbaki, Church, notamment.

39 D. MARR, « Selections from Vision », dans A. Noë & E. Thompson (dir.), *Vision and Mind. Selected Readings in the Philosophy of Perception*, Cambridge, The MIT Press, 2002, p. 244. Je traduis : « To say that something is a formal scheme means only that it is a set of symbols with rules for putting them together—no more no less. »

40 Bien qu'un langage formel exige des symboles discrets et atomiques et des règles strictes de leur manipulation, il ne s'ensuit pas que ces langages formels sont incapables de traiter du continu, du parallèle et de l'infini. Ceci relèvera de leur sémantique.

41 À titre d'exemple simple, une certaine relation d'implication entre des propositions *p* et *q* sera dans le formalisme russellien écrit $p \longrightarrow (p \longrightarrow q \vee z)$ alors que, dans la notation polonaise, elle devient *CpAqz* (cf. J. LUKASIEWICZ, *Elements of Mathematical Logic*, New York, MacMillan, 1963 [1929]). Mais deux formalismes ou notations exprimeront la même structure d'implication logique.

42 Pour Suárez et Solé, le rôle de la représentation comme mise en relation entre un objet d'étude et un modèle ne peut être le seul rôle d'un modèle. Il doit permettre des inférences. M. SUÁREZ & A. SOLÉ, « On the Analogy between Cognitive Representation and Truth », *Theoria*, vol. 21, no 1, 2006, p. 39. En ligne : <https://www.ehu.eus/ojs/index.php/THEORIA/article/view/552/401>. Je traduis : « In a scientific representation some source A – typically a model, a graph, an equation – is used to represent some target B – typically a system, entity or phenomenon. »

43 La sémantique de ces symboles ne se limite évidemment pas à la question de la référence et la signification n'épuise pas le « sens ».

44 Dans un langage logique, une constante peut se référer à une personne et un prédicat à une propriété, par exemple la phrase formelle logique suivante « S(j) et B(j) » peut être interprétée de la manière suivante : « j » se réfère à « John », « S » à la propriété « Student » et « B » à la propriété « Brilliant ». Ce n'est donc pas parce qu'un modèle est formel qu'il fait nécessairement référence à des nombres ou, comme on le dit, qu'il est « quantitatif ». Dire que les symboles formels sont relatifs à des quantités est une proposition ambiguë : tous les symboles formels ne sont pas des chiffres. L'École de Bourbaki refusera précisément de voir la logique ou l'arithmétique comme le domaine premier de la mathématique.

45 Mahoney écrit par exemple : « Science is not about nature, but about how we represent nature to ourselves. » M. S. MAHONEY, « The Structures of Computation and the Mathematical Structure of Nature », *The Rutherford Journal*, vol. 3, 2010. En ligne : <http://www.rutherfordjournal.org/article030107.html>.

46 N. BOURBAKI, *Théorie des ensembles*, Berlin, Springer, 2006 [1970].

47 Ceci est un point important de la différence entre un langage ou langue formelle et un langage naturel. Certaines théories sémantiques soutiennent que la signification et le sens émergent des intentions de communication des locuteurs, plusieurs théories sémantiques soutiennent par ailleurs que la langue participe profondément à l'organisation et à la structuration de la sémantique d'un système formel.

48 Ici tant l'épistémologie sociale que la sociologie de la science peuvent jouer un rôle important dans l'explicitation du rôle épistémique du « créateur » de ces modèles. Ce « créateur » n'est souvent pas nécessairement un individu. Il peut être une communauté épistémique qui n'est pas sans être influencée par d'autres théories épistémiques antérieures ou périphériques, mais aussi des conditions sociopolitiques de production.

49 M. S. MORGAN & M. MORRISON (dir.), *Models as Mediators, op. cit.*

50 G. ISRAEL, *La Mathématisation du réel. Essai sur la modélisation mathématique*, Paris, Seuil, 1996, p. 75.

51 Il faut se rappeler que le symbole x de la multiplication est une abréviation qui réfère à une sommation complexe de la valeur d'un nombre. Ainsi 3 x 3 est une abréviation (3 + 3) + (3 + 3) + (3 + 3) = 9.

52 D. HILBERT, *Les fondements de la géométrie*, trad. de l'allemand par P. Rossier, Paris, Dunod, 1971 [1899].

53 A. M. TURING, « On Computable Numbers, with an Application to the Entscheidungsproblem », *Proceedings of the London Mathematical Society*, vol. 42, no 1, 1937, p. 230-265.

54 A. CHURCH, « An Unsolvable Problem of Elementary Number Theory », *American Journal of Mathematics,* vol. 58, no 2, 1936, p. 345-363.

55 H. B. CURRY & R. FEYS, *Combinatory Logic. Volume 1*, Amsterdam, North-Holland Publishing Company, 1958.

56 A. MARKOV, « The Theory of Algorithms », *American Mathematical Society Translations: Series 2*, vol. 15, 1960, p. 1-14.

57 E. L. POST, « Recursively Enumerable Sets of Positive Integers and their Decision Problems », *Bulletin of the American Mathematical Society*, vol. 50, no 5, 1944, p. 284-316. En ligne : <https://projecteuclid.org/download/pdf_1/euclid.bams/1183505800>.

58 R. GANDY, « Church's Thesis and Principles for Mechanisms », *Studies in Logic and the Foundations of Mathematics*, vol. 101, « The Kleene Symposium », 1980, p. 123-148.

59 J. von NEUMANN, « The general and logical theory of automata (1951) », dans *John von Neumann - Collected Works*, vol. 5, textes rassemblés par A. H. Taub, Oxford, Pergamon Press, 1961, p. 303. Je traduis : « The theory of automata [...] will have to be, from the mathematical point of view, combinatorial rather than analytical. »

60 N. CHOMSKY, « On certain formal properties of grammars », *Information and Control*, vol. 2, no 2, 1959, p. 137-167.

61 P. J. LANDIN, « The Next 700 Programming Languages », *Communications of the ACM*, vol. 9, no 3, 1966, p. 157-166.

62 N. CHOMSKY, « Trois modèles de description du langage », *Langages*, no 9, 1968, p. 52 (traduction de N. CHOMSKY, « Three Models for the Description of Language », *IRE Transactions on Information Theory*, vol. 2, no 3, 1956, p. 113-124).

63 M.-P. SCHÜTZENBERGER, « Some Remarks on Chomsky's Context-Free Languages », *Quarterly Progress Report of the MIT Research Laboratory of Electronics*, no 68, 1961, p. 155-170.

64 Ces travaux reçurent beaucoup d'échos en France par le biais des travaux de Maurice Gross et André Lentin (cf. M. GROSS & A. LENTIN, *Notions sur les grammaires formelles*, Paris, Gauthier-Villars, 1970) ainsi que ceux des mathématiciens linguistes du Centre des mathématiques appliquées (cf. J.-P. DESCLÉS, « La Grammaire Applicative et Cognitive construit-elle des représentations universelles? », *loc. cit.*). Ces travaux donnèrent une rigueur mathématique au structuralisme naissant.

65 D. LACOMBE, « La théorie des fonctions récursives et ses applications », *Bulletin de la Société Mathématique de France*, tome 88, 1960, p. 393-468.

66 J. McCARTHY, « Recursive Functions of Symbolic Expressions and their Computation by Machine – Part 1 », *Communications of the ACM*, vol. 3, no 4, 1960, p. 184-195 ; J. McCARTHY, « A Basis for a Mathematical Theory of Computation », *Proceedings of the Western Joint Computer Conference*, 1961, p. 225-238.

67 C. STRACHEY, « Towards a Formal Semantics », dans T. B. Steel (dir.), *Formal Language Description Languages for Computer Programming*, Amsterdam, North-Holland Publishing Company, 1966, p. 198-220.

68 R. M. BURSTALL & P. J. LANDIN, « Programs and Their Proofs: An Algebraic Approach », *Machine Intelligence*, vol. 4, 1969, p. 17. Je traduis : « Programming is essentially about certain "data structures" and functions between them. »

69 A. ASPERTI & G. LONGO, *Categories, Types and Structures*, Cambridge, The MIT Press, 1991.

70 R. M. BURSTALL & P. J. LANDIN, « Programs and Their Proofs: An Algebraic Approach », *loc. cit.*, p. 170. Je traduis : « Programming is essentially about certain "data structures" and functions between them. [...] Starting with such familiar algebraic structures as groups and rings algebraists have developed a wider notion of algebraic structure (or "algebra") which includes these as examples and also includes many of the entities which in the computer context are thought of as data structures. »

71 J. R. BÜCHI, « Algebraic Theory of Feedback in Discrete Systems, Part I (1966) », dans *The Collected Works of J. Richard Büchi*, textes rassemblés par S. Mac Lane & D. Siefkes, New York, Springer, 1990, p. 339. Je traduis : « If the definition of "finite automaton" is appropriately chosen, it turns out that all the basic concepts and results concerning structure and behavior of finite automata are in fact just special cases of the fundamental concepts (homomorphism, congruence relation, free algebra) and facts of abstract algebra. Automata theory is simply the theory of universal algebra [...] with unary operations, and with emphasis on finite algebras. »

72 D. SCOTT, « Outline of a Mathematical Theory of Computation », *Technical Monograph PRG-2*, Oxford University Computing Laboratory, 1970, p. 170. Je traduis : « The main mathematical novelty of the present study is the creation of a proper mathematical theory of functions which accomplishes these aims (consistently!) [...] »

73 M. S. MAHONEY, « The Structures of Computation and the Mathematical Structure of Nature », *loc. cit.*, non paginé. Je traduis : « In the absence of mathematical structures that allow abstraction and generalization, computational models do not say much. Nor do they function as models traditionally have done in providing an understanding of nature on the basis of which we can test our knowledge by making things happen in the world. »

74 S. GREEN, « When one Model is not Enough: Combining Epistemic Tools in Systems Biology », *Studies in History and Philosophy of Biological and Biomedical Sciences*, vol. 44, no 2, 2013, p. 170-180.

75 J. LAMBEK & M. LAMBEK, « The Kinship Terminology of Malagasy Speakers in Mayotte », *loc. cit*.

76 Ce terme d'idéalisation est un des rôles épistémiques que plusieurs théoriciens donnent aux modèles scientifiques. Cf. M. S. MORGAN & M. MORRISON (dir.), *Models as Mediators*, *op. cit.*

77 Il est théoriquement possible de représenter des évènements dynamiques par des propositions logiques. Mais cela exigera des symboles qui puissent représenter chacun des instants-moments de l'évolution d'un phénomène et donc ultimement de simuler le continu. Cela rendra lourd le système logique. Et on imagine comment il sera difficile de faire pour des systèmes de type dynamique et chaotique. Même les logiques du temps doivent se restreindre dans les objets dynamiques qu'ils peuvent représenter. Ce n'est pas sans raison que les pratiques scientifiques ne représentent pas les phénomènes dynamiques par le biais de la logique des prédicats, des propositions et même des logiques modales.

78 Voir la critique de Piotrowski et Visetti sur l'application de ces types de modèles a la sémiolinguistique. D. PIOTROWSKI & Y.-M. VISETTI, « Connaissance sémiotique et mathématisation. Sémiogenèse et explicitation », *loc. cit*.

79 K. GÖDEL, *On Formally Undecidable Propositions of Principia Mathematica and Related Systems*, trad. de l'allemand par B. Meltzer, New York, Dover Publications, 1992 [1931].

80 G. CHAITIN, F. A. DORIA & N. DA COSTA, *Gödel's Way: Exploits into an Undecidable World*, Boca Raton, CRC Press, 2012.

81 M. DAVIS, *Computability and Unsolvability*, New York, Dover Publications, 1982.