AS SA *Applied Semiotics / Sémiotique appliquée*, nº 26 (2018)

# VERS UNE SÉMIOTIQUE COMPUTATIONNELLE ?

Jean-Guy Meunier
Université du Québec à Montréal
meunier.jean-guy@uqam.ca

**Résumé**
La sémiotique est une théorie des signes. Un des objectifs de recherche régulièrement réaffirmée par les sémioticiens est la manipulation et l'interprétation de ces signes. Il s'avère que des fondateurs (Newell et Simon, 1976) de l'intelligence artificielle ont aussi défini leur discipline comme la science de la manipulation des signes et plus particulièrement des symboles ! Certains sémioticiens ont alors proposé de définir l'ordinateur comme une machine sémiotique. Certains autres chercheurs ont proposé de définir la manipulation des signes comme une forme de la computation. La recherche poursuivie dans cet article explore le lien entre les notions de sémiotique et de computation. Elle voit la sémiotique comme une démarche épistémique qui construit des modèles. Certains de ceux-ci peuvent être de type computationnel. Elle analyse aussi la portée, mais aussi les limites, de ce lien entre sémiotique et computation.

## Introduction : sémiotique et formalisation

La sémiotique est un type de recherche qui s'intéresse à l'analyse, la description et l'explication de réalités sémiotiques, c'est-à-dire des objets, faits ou actions qui dans le monde sont porteurs de significations pour des humains. En termes classiques, pour reprendre la définition de Peirce, elle est une forme de logique des signes : « *Logic is [...] synonymous with semeiotic, the pure theory of signs in general.* » (Peirce, s.d.)

Mais cette théorie des signes n'est pas des plus faciles à construire. Divers courants la traversent. Pour notre propos, qui s'intéresse ici aux liens entre sémiotique et computation, nous nous attarderons plus spécifiquement à deux des plus importants courants.

Un premier s'intéressera avant tout aux fonctions et usages des signes. Les signes ne sont pas par eux-mêmes porteurs de signification. Ils la reçoivent dans la parole (Saussure), la communication (Jakobson, Martinet, Halliday, Eco, etc.) ou le dialogue (Habermas, Apel, Ricœur) ou l'intentionnalité (Austin, Searle, Grice) que les agents leur assignent. Et les théories du langage initiées par des Herder, Hermann, Humboldt et évidemment Wittgenstein, bien que ne s'exprimant pas toujours en termes de sémiotique, insisteront sur l'usage effectif des signes pour une théorie de la signification. Ce courant est très riche et a ouvert des voies de recherche des plus fécondes. Cependant, rares sont les projets qui se sont aventurés dans une certaine formalisation de la démarche d'analyse sémiotique. Ils sont encore plus loin d'un lien avec la computation. On dirigera plutôt la description et l’explication sur les multiples conditions de participation à la communication et l’usage des signes.

Le second courant s’intéressera à l’organisation et la structuration des signes eux-mêmes indépendamment de leur usage et communication. Deux traditions scientifiques en proposeront la compréhension. Dans une première — le structuralisme d’influence saussurien (Saussure, Hjemslev, Jakobson, Greimas, Rastier, etc.) – on soutiendra qu’une théorie des signes ne peut avoir comme objet des signes isolés. Elle doit s’intéresser à leur organisation ou à leur structure : c’est-à-dire, elle doit identifier, décrire et expliquer les unités et les relations différentielles qui les construisent.

L’exemple prototypique de systèmes de signes sera la langue. Pour Saussure (1990 : 159), « dans la langue tout repose sur des rapports ». La langue est un « système dont tous les termes sont solidaires et où la valeur de l’un ne résulte que de la présence simultanée des autres ». Elle est régie par une multiplicité de types de relations (syntagmatiques, associatives, paradigmatiques, oppositionnelles, etc.) et, en conséquence, elle peut être comprise comme un système formel *:* « la langue est pour ainsi dire une algèbre qui n’aurait que des termes complexes » (Saussure, 1990 : 168). Une tout autre tradition – le logicisme (Frege, Carnap, Morris, Ogden et Richard, etc.) – soutiendra qu’une théorie des signes doit décrire et expliquer les conditions nécessaires de leur structuration (syntaxe), de leur fonction signifiante (sémantique) et de leur usage (pragmatique). La sémiotique est alors une théorie des systèmes symboliques. Ceux-ci sont constitués de symboles et de règles qui permettent de composer des configurations (dites formules) simples ou complexes de signes ou symboles. Les langues logiques en sont le prototype. Bref, tant le structuralisme que le logicisme soutiendront la thèse qu’une théorie sémiotique adéquate et riche doit décrire et expliquer non pas les signes isolés, mais leur organisation et leur structuration. Reste évidemment à déterminer la manière et la méthode pour ce faire.

On trouvera ainsi des exemples riches et féconds de l’application de ces hypothèses structuralistes et logicistes à divers systèmes de signes. La langue sera certes le domaine prototypique de son application. Mais on l’explorera aussi dans de nombreux autres types d’artefacts sémiotiques dont on proposera de formaliser certaines dimensions. Pensons aux analyses des contes et des récits (Propp), de la langue (Hjemslev), des romans (Greimas), de la culture (Lotman), du discours (Halliday), des films (Metz), des mythes (Lévi-Strauss), de la communication (Andersen), du marketing (Sebeok), de la musique (Nattiez, Jackendoff), du visuel (Groupe µ, Saint-Martin), de la cognition (Brandt), etc.

Rares cependant seront les analyses qui en appelleront systématiquement à une formalisation rigoureuse et complète. Harris (1968) et Sebeok (1976) seront parmi ceux, qui, à l’instar de Carnap et Morris (1971), tenteront de dégager des principes formels d’une analyse sémiotique formelle. Pelc (1979) et Lieb (1979) proposeront des modèles d’une sémiotique formelle, Marty (1979) lira Peirce à travers la théorie des catégories, Petofi (1978), van Dijk (1978) et Pavel (1975) proposeront des grammaires formelles pour l’analyse des textes. Lambek et Lambek (1981) appliqueront la logique catégorielle à l’analyse de relations de parenté, Jackendoff (1987) décrira la musique de manière générative. Petitot (1985) proposera des analyses sémiotiques au moyen de la théorie de morphodynamique de

Thom. Desclés (1990) en appellera à la logique combinatoire pour la description sémiotique du langage naturel. Tijus et ses collègues (2006) appliqueront des grammaires catégorielles à des pictogrammes. Hoffmeyer et Favereau (2009) liront la biologie à travers la sémiotique, Vickers et coll. (2013) et Nowakowska (1981) proposeront des sémiotiques formelles pour la visualisation et les systèmes multidimensionnels.

**1. La « sémiotique computationnelle »**

Malgré la richesse et la pertinence de ces hypothèses formalistes ou structuralistes et de ces programmes de recherche, peu de ces chercheurs feront un lien entre cette formalisation et la computation. Ce n'est que récemment, depuis une vingtaine d'années, qu'apparaitra un courant de recherche sémiotique qui fera explicitement ce lien avec le computationnel et qui s'appellera, en français, sémiotique computationnelle ou en anglais, *computational semiotics*. Dans ce courant, on peut distinguer deux programmes différents de recherche.

Un premier programme voit la computation elle-même comme une forme de théorie sémiotique. Ce que l'on vulgarise souvent en disant que l'ordinateur est un artefact sémiotique parmi d'autres. Comme le dira Nadin (1977, 2007), l'ordinateur est le prototype d'une machine sémiotique : « Pour être signifiant les ordinateurs doivent être des machines sémiotiques [1].» Une telle vision traversera plusieurs projets de type intelligence artificielle : par exemple, la communication humains-machine (De Sousa, 2005), la programmation (Tanaka-Ishii, 2010), le design informatique (Liu, 2005), et même la robotique (Meystel et Albus, 1991). Floridi (1999) est peut-être celui qui a le plus dynamisé l'importance de la vision informationnelle et sémiotique de l'informatique.

Un second programme inversera la relation entre la sémiotique et le computationnel. Cette fois, on cherchera à expliquer des faits ou formes sémiotiques par la computation. Ainsi, la théorie sémiotique inclurait dans ses outils méthodologiques la notion de computation. Ce qui serait vulgarisé ainsi : certains artefacts sémiotiques peuvent être traités par des algorithmes.

Une telle perspective sera exploitée dans plusieurs projets de type sémiotique. On la retrouvera très ancrée en sciences cognitives classiques qui, malgré un vocabulaire non sémiotique, soutiendra, à l'instar de Fodor (1975), que l'esprit [*mind*] peut être compris comme un système computationnel de représentations, c'est-à-dire en termes sémiotiques : de quelque chose qui se tient à la place de… (*aliquid stat pro aliquo*) ou encore en termes plus contemporains de signes/signaux/informations.

Une même vision se retrouvera aussi dans les sciences humaines, les lettres, la linguistique, la communication, les médias. Certains artefacts sémiotiques de ces disciplines seront décrits et expliqués par des concepts et des théories de nature computationnelle.

[1] « *[I]n order to be meaningful, computers ought to be semiotic machines* » (Nadin, 1977). Nous traduisons.

Plusieurs projets de recherche s'inscriront dans ce programme : par exemple, dès les années 1970, plusieurs chercheurs tant en Europe qu'en Amérique exploreront divers domaines par des approches linguistiques et mathématiques (Busa, 2008 ; McKinnon, 1973 ; Brunet, 1986 ; Lebart et Salem, 1988 ; Bolasco et Salem,1996 ; Hockey, 2000 ; McCarthy, 2008). Ce n'est que plus tard que certains de ces territoires seront approchés de manière ouvertement sémiotique. Pensons, par exemple à l'analyse sémio-computationnelle du langage naturel (Desclés, 1997), l'analyse sémantique (Rieger, 1999), la communication personne-machine (Andersen, 1991), l'herméneutique matérielle (Bachimont, 1996 ; Rastier, 2011 ; Mayaffre, 2000), mais plus récemment, et avec un dynamisme surprenant : les humanités numériques (McCarthy, 2008 ; Hockey, 2000 ; Rockwell, 2003 ; Douelhi, 2011 ; Dacos et Mounier, 2014 ; Meunier, 2015 ; etc.). Dans ce programme de recherche, la thèse principale sera que les signes et symboles qu'étudie la sémiotique peuvent être traités et analysés de manière computationnelle.

Dans ces deux programmes, le lien entre sémiotique et computation est direct. De plus, la computation elle-même est aussi intimement liée à l'informatique. Il reste cependant qu'à part les humanités numériques, cette sémiotique computationnelle, comme telle, n'a pas réussi véritablement à prendre son envol. Elle est demeurée relativement marginale par rapport aux principaux courants de la sémiotique. En outre, elle a été souvent soumise à des critiques importantes.

Ainsi, relativement au premier programme de recherche, certains douteront de la pertinence des concepts sémiotiques pour fonder les projets informatiques (Pylyshyn, 1984), non pas en raison de sa fausseté, mais en raison de sa trop grande généralité et de son manque de rigueur dans sa conceptualité et ses formalisations. Et plus généralement, les informaticiens trouveront non pertinent ou inutile de penser la théorie de computation comme une forme de théorie sémiotique. Cette difficulté pour la sémiotique d'en appeler à quelconque formalisation lui vaudra des critiques sévères. On ne la verra pas comme une démarche scientifique crédible. Elle apparaîtra plutôt comme une forme plus ou moins rigoureuse de démarche interprétative portant sur des artefacts sémiotiques.

Le deuxième programme de recherche rencontrera aussi une résistance. Plusieurs sémioticiens refuseront d'admettre que les problèmes sémiotiques complexes comme ceux que l'on trouve dans les arts, les lettres, le social, etc., puissent être décrits et expliqués de manière computationnelle. On verra ce programme de recherche comme reprenant la théorie positiviste des automates ou de la cybernétique. Pire encore : l'ordinateur ne sera qu'un gadget dans une démarche sémiotique.

La compréhension de ce lien entre sémiotique et computation n'est pas une problématique simple. Ce lien existe, mais il est plus complexe que la formulation vulgarisée qui le traduit en termes de relation entre *ordinateur* et *sémiotique*.

## 2. La sémiotique comme une science modélisatrice

Pour explorer cette problématique, nous commencerons par nous interroger sur la nature de la pratique scientifique en jeu dans la sémiotique pour mieux voir comment et où elle pourrait être liée à la computation. Pour ce faire cependant, il faut préciser ce qu'est cette scientificité de la sémiotique. Car, dès le point de départ, plusieurs refuseront de voir la sémiotique comme une science, c'est-à-dire une science comme le voit l'épistémologie classique de type logico-positiviste, à savoir : une pratique discursive qui s'exprime dans un langage formel et valide ses propositions par de l'expérimentation.

Une telle compréhension de la pratique scientifique est évidemment problématique pour la sémiotique. Elle relève, on le sait, d'une vision de la science tout à fait spécifique, c'est-à-dire, où la science est avant tout une entreprise de formalisation nomologique et logique comme le proposent Schlick (1979), Neurath (1973), Carnap (1937), Hempel (1965) ou Nagel (1961). D'autre part, même si ces types de formalisation étaient pertinentes et utiles pour une modélisation formelle, rien n'assurerait qu'elles aient la puissance symbolique adéquate pour traiter la complexité des certains phénomènes tels le climat, l'astrophysique, les réseaux sociaux, etc. Et encore moins les artefacts sémiotiques.

Cette vision de la science, on le sait, n'en est qu'une parmi plusieurs. Elle a été des plus critiquées. En sciences sociales et humaines, elle fut régulièrement remise en question tant par l'École de Francfort et plus particulièrement Horkheimer (1974). On trouvera trop réductrice cette vision syntaxique et objectiviste de la science. Elle est peu pertinente dans les sciences humaines, qui sont confrontées à de l'interprétation complexe et qui déterminent ultimement la référence et la vérité des propositions. Interprétation qui se construit non pas par stipulation, mais dans l'usage, le dialogue et la communication. La perspective doit être plus herméneutique (Habermas, 1973).

Cependant, entre un vison positiviste et une vision herméneutique, une autre vision de la science émergera avec les travaux de Kuhn (1962) et Feyerabend (1975). Celle-ci cherchera à retenir certains éléments de l'une et l'autre de ces deux visions précédentes. Ici, la science sera comprise avant tout comme une démarche de connaissance. Pour un Kuhn, la science est une démarche cognitive dont les propositions ou les croyances sont partagées (des paradigmes) par une communauté épistémique. Pour Bachelard (1979), Canguilhem (1977), van Fraassen (2008) et Thagard (2004), elle sera une construction, reconstruction et combinaison de représentations épistémiques souvent nombreuses et diversifiées qui, chacune à sa manière, joue un rôle particulier dans la création des concepts et des théories. Plus récemment, Cartwright (1983), Giere (1988, 1999), Rheinberger (1997), Leonelli (2007), Morgan et Morrison, (1999) et Frigg (2011) considéreront comme trop générale cette notion de représentation et la retraduiront en termes de *modèle*.

L'aspect original de cette vision est que les théories que la science construit ne sont pas définies comme un ensemble fixe d'énoncés (axiomes, théorèmes, etc.) comme le proposait Carnap, mais plutôt comme un ensemble de modèles qui émergent, se croisent, se complètent, se stabilisent, s'affaiblissent et finalement meurent. Ainsi, une théorie scientifique est une dynamique cognitive complexe fondée sur une multiplicité de modèles interreliés.

Ces modèles sont des artefacts épistémiques qui participent à la description, l'explication et la compréhension des objets de recherche. Ils présentent cependant une propriété commune : ils s'expriment tous dans des formes sémiotiques, plus particulièrement, des langages ou des systèmes de symboles. Certains d'entre eux sont non formels (langue naturelle, langage iconique), d'autres sont strictement formels (algèbre, géométrie, logique, algorithmes), d'autres enfin sont hybrides (un mixte de langage naturel et de langage iconique et formel, etc).

Chacun de ces modèles aura des rôles épistémiques spécifiques qui, ultimement, viseront la description et l'explication des phénomènes. Mais tous seront interreliés d'une manière ou d'une autre et ensemble, ils seront les porteurs des concepts et des théories scientifiques. La richesse et la diversité de ces modèles enrichiront la démarche scientifique.

Cette vision de la science est des plus heuristiques pour comprendre la démarche sémiotique. Celle-ci peut ainsi être vue comme une pratique épistémique spécifique qui, à sa manière, construit un ensemble de modèles descriptifs et explicatifs de phénomènes ou de réalités d'un type particulier : *les artefacts porteurs de significations*. En termes plus simples, la sémiotique est une activité cognitive de compréhension modélisatrice d'artefacts signifiants. Et comme toute science, elle construira nécessairement plusieurs modèles. Mais vu la complexité des objets qui intéressent la sémiotique, certains modèles seront privilégiés pour construire une théorie.

## 3. La variété des types de modèles

Plusieurs chercheurs ont étudié des pratiques effectives en science (Leonelli, 2007 ; Knuuttila et Loettgers, 2012). Ils ont montré comment, dans ces pratiques, les théories scientifiques sont peuplées d'une multitude et d'une diversité de modèles.

Dans certaines sciences, les modèles créés ont des rôles cognitifs différents, quoique pas toujours précis, exclusifs et bien identifiés. Certains servent à l'exploration des intuitions, des croyances, des hypothèses qu'il faut préciser, ajuster, définir, etc. D'autres, au contraire, reposent sur des expertises bien maitrisées ou des protocoles standardisés d'expérimentation que l'on doit adapter, ajuster, reformuler, etc. Certains jouent un rôle dans la démonstration, le prototypage, la simulation, etc. D'autres prennent des formes symboliques complexes de type algébriques, mécaniques, graphiques, systémiques, etc. Enfin, d'autres sont évaluatifs, critiques, révisionnistes, etc.

Bien que certains modèles soient plus valorisés ou importants que d'autres, aucun n'est hégémonique par rapport aux autres. Et souvent, seuls quelques-uns répondront aux canons épistémologiques de types logico-nomologiques. Autrement dit, une théorie scientifique met en œuvre une dynamique épistémique complexe où les modèles contribuent, individuellement et collectivement, à l'émergence d'un savoir. Dans une telle perspective, par la configuration de ses modèles, chaque science se construit une signature propre.

Notre intuition ici est que la sémiotique est aussi une pratique modélisatrice pluraliste. Devant la complexité de certains artefacts, faits et événements, une théorie sémiotique adéquate ne peut que construire une multiplicité de modèles. Selon son rythme d'évolution, de développement et à certains moments de son développement, certains modèles seront privilégiés et d'autres seront mis en retrait. Mais aucune recherche sémiotique ne peut penser n'avoir qu'un seul type de modèles et s'y restreindre. Une recherche sémiotique sérieuse exige, comme toute autre science, divers points de vue, angles d'approche, expertises, postures, etc. Chaque projet sémiotique contiendra ses ensembles propres de modèles. Par exemple, un littéraire qui approche le fait textuel n'aura pas les mêmes modèles que l'anthropologue qui scrute des rituels, pas plus que le sociologue qui étudie les représentations sociales ou le politicologue qui analyse des discours d'un président.

Dans une telle perspective d'ouverture, certains pensent que la recherche sémiotique peut introduire, parmi plusieurs autres, des modèles formels et ultimement des modèles computationnels pour enfin créer des modèles informatiques.

Ces trois types de modèles entretiennent de fait des liens étroits. Et comme nous le montrerons dans la présente recherche, il ne peut y avoir de modèle computationnel que s'il existe au préalable des modèles formels. Et ce n'est qu'à cette condition qu'une modélisation informatique est possible. Par ailleurs, l'existence d'un modèle formel ne garantit pas celle d'un modèle computationnel et celle d'un modèle informatique. Nous étudierons maintenant ces trois types de modèles.

## 4. Le modèle formel

Nous commençons par le modèle formel. Pour la tradition épistémologique classique, un modèle est dit formel si la forme de son expression contient un ensemble fini de symboles régis par des règles strictes de composition. Par commodité, on appellera souvent ces systèmes symboliques formels des *langues formelles*[2] où les symboles de départ constituent un vocabulaire et les règles de syntaxe permettent de générer et de transformer des formules. Cette langue formelle est sans sémantique[3]. On distinguera des modèles formels selon le type de systèmes symboliques qu'ils utilisent.

Un premier type de modèles formels est axiomatique[4]. Il est le modèle idéal proposé par la philosophie de sciences classiques (Carnap, 1935 ; Hempel, 1965 ; Suppes, 1967). Ce

[2] En anglais, on utilisera le terme « *language* » pour « langue ». Mais en français, le terme « langage » couvre plus que la « langue ». Par exemple, on peut voir les règles de conversation comme des normes dans le langage, mais non des règles d'une langue. Et ce terme lui-même est problématique, car il existe des systèmes symboliques qui ne sont pas des langues, au sens strict de ce terme. Par exemple, les systèmes symboliques mathématiques ou encore des systèmes symboliques contenant des formes graphiques comme + *, des lignes, des cercles, des figures (graphes, plans, etc.).

[3] Cependant, un modèle formel sera dit interprété si, dans une application à un domaine spécifique, une sémantique lui est ajoutée.

[4] On peut définir un système axiomatique comme un ensemble de formules symboliques liées par des liens déductifs réglés et dont certaines des formules sont des axiomes et les autres, des théorèmes. Un système

type de langage possède des règles particulières de transformation de ses formules (inférence, substitution, etc.). Certaines formules sont choisies comme postulats, axiomes, théorèmes, lemmes, corolaires, etc. Dans certaines applications scientifiques, des formules seront considérées comme des lois. Des séquences de formules serviront de démonstration et de preuve. Les prototypes de systèmes axiomatiques sont les divers systèmes logiques : logique propositionnelle, prédicative, combinatoire, etc. Dans ces systèmes, la sémantique joue un rôle secondaire. On les utilise surtout pour modéliser des structures mathématiques. Peu de sciences l'utiliseront pour la modélisation de leur objet de recherche. Ils se sont avérés loin de la pratique concrète des sciences.

Le deuxième type de modèles formels est mathématique. Dans la pratique, ces modèles sont les plus utilisés en science. Ils possèdent aussi un vocabulaire et une syntaxe. Ils se caractérisent, entre autres, par l'inclusion de classes de symboles (constantes, variables, opérations, etc.). Ils ajoutent aussi des types particuliers de règles pertinentes à des manipulations complexes[5] de leurs symboles (substitution, élimination, récursion, transformation, etc.).

Cependant, certaines formules seront importantes pour la description et l'explication des phénomènes, entre autres : les formules qui expriment des relations fonctionnelles[6], communément appelées des fonctions.

Par exemple, si la langue formelle est de type algébrique, la grammaire de cette langue formelle permettra de générer une formule (équation) qui exprime une relation fonctionnelle : $F(x, y){:}\, z = (x + y)^2$ et à partir de laquelle, en suivant des règles, on pourra générer l'équation $F(x, y){:}\, z = x^2 + 2xy + y^2$, peu importe l'univers de référence auquel renvoie cette équation.

Les prototypes de ces systèmes symboliques mathématiques sont l'arithmétique, les algèbres, les statistiques, la géométrie, le calcul lambda, les théories des ensembles, des groupes, des graphes, du chaos, les grammaires catégorielles, etc.

Ces systèmes symboliques sont aussi par définition sans sémantique. Cependant, dans leur utilisation concrète dans une science, ils recevront une sémantique spécifique, construisant ainsi un modèle formel mathématique interprété. Ainsi, ce modèle reliera le système symbolique à divers types d'entités et de relations du domaine étudié. Par exemple, en physique, l'équation $F = ma$ est une formule syntaxiquement bien faite et où les symboles renvoient à des entités d'un monde étudié par les physiciens. Ainsi, la vérité des formules dépendra du domaine d'application, c'est-à-dire de la sémantique qui lui est assignée.

---

axiomatique hilbertien présente des propriétés spécifiques. Il doit avoir une description finie. Il est un système construit à partir d'un nombre fini d'axiomes et de règles d'inférence.

[5] Dans la majorité de leur utilisation effective, ils ne définiront pas explicitement leurs axiomes ou postulats. Certains cependant le feront, par exemple : l'axiomatisation de l'arithmétique de Peano.

[6] Il peut exister d'autres types de modèles formels et qui ne sont pas fonctionnels. En effet, on peut construire un modèle qui s'intéresse à des relations non fonctionnelles entre éléments, par exemple à des réseaux de personnes dans un groupe. Le modèle sera relationnel. Mais ces types de relations ne seront pas ceux qui sont retenus dans un modèle formel fonctionnel.

Une conséquence immédiate est qu'un modèle formel ne porte pas nécessairement sur du quantitatif. Aussi, on trouvera des formalismes logiques, géométriques, topologiques, grammaticaux, etc. Certains utiliseront des symboles iconiques (graphes, images, etc.). Dans tous ces formalismes, on pourra trouver divers types de symboles : tels des constantes, des variables, des opérateurs, etc.

Ces modèles formels mathématiques sont omniprésents tant dans les sciences naturelles que dans les sciences humaines et sociales. Par exemple, on retrouve ces types de modèles formels en économie qui, pour expliquer des dynamiques de marché, en appellera à la théorie des jeux, aux systèmes d'équilibre, au calcul d'optimisation, aux probabilités bayésiennes, etc. Les neurosciences exploreront des modèles formels d'algèbre non linéaire de classification, de catégorisation et qu'ils nommeront métaphoriquement des réseaux de neurones. L'archéologie contemporaine utilisera simultanément ou séquentiellement des modèles statistiques, géométriques, graphiques, algébriques, linguistiques qui, interactivement, participeront à la description de la fouille d'un site.

On retrouve aussi des modelés formels en sémiotique. Cependant, peu sont de types arithmétiques, ou algébriques, mais certains sont de type grammatical ou logique. Le carré logique ; les grammaires génératives, catégorielles, les graphes, en sont des exemples classiques.

Nous avons insisté dans cette section sur des détails un peu techniques sur la nature d'un modèle formel. Ces détails sont essentiels pour comprendre le second type de modèles que nous aborderons maintenant : le modèle computationnel. Car le modèle formel est la porte incontournable à la computationnalité. Sans lui, il n'y aura pas de computation possible. Malgré son omniprésence en science, ce type modèle pose un type problème important : la validité de ses formules. Ce problème invite à la création d'un second type de modèles : le modèle formel computationnel.

## 5. Les modèles computationnels

Le deuxième type de modèles est computationnel. Ces modèles sont des variantes d'un modèle formel. Ils sont un sous-ensemble des modèles formels mathématiques. Comme ces derniers, les modèles computationnels utilisent des systèmes symboliques, mais dont les formules ont une propriété tout à fait particulière à savoir : être « computables ». Nous expliciterons un peu plus loin la nature de cette propriété. Comme nous le verrons, cette propriété n'est pas simple. Depuis des siècles, elle est au cœur d'interrogations très techniques des mathématiques. Et aujourd'hui, elle sert de fondement théorique aux technologies informatiques. De plus, elle n'est pas sans lien avec la sémiotique.

Sur le plan historique, cette question de la computation est ancienne. Déjà au 13$^{e}$ siècle, le logicien John de Salisbury (1115-1180) évoquait dans ses écrits qu'un de ses étudiants, William de Soissons, avait construit une *machine* « machina » (*engine*) capable de démolir des théories logiques et de construire des conclusions à partir de principes logiques à l'aide d'une *machine* capable par une *manipulation de symboles* de montrer la *validité* d'un raisonnement.

Une même idée se retrouvera un siècle plus tard chez Ramon Lull (1232-1315) qui proposait un artefact en papier constitué de cercles imbriqués contenant des *symboles* discrets (représentant des concepts) et sur lequel on pouvait appliquer des manipulations physiques de triangles ou de leviers (voir la figure 1). Cet artefact permettait d'effectuer la démonstration que certaines configurations de symboles étaient vraies ou fausses. Cet artefact fut historiquement interprété comme une « machine ». Selon (Kneale et Kneale, 1988), elle serait le premier modèle des machines à « computer ».

*La roue logique de Lull*

Figure 1

L'intérêt de cette « machine » repose sur deux propriétés importantes qui seront reprises plus tard dans les théories de la computation. Premièrement, cette « machine » qui effectue la computation est une technologie physique (une roue faite d'un matériau particulier), sur laquelle sont inscrits des signes discrets et qui, par une manipulation de triangles ou de leviers, permet une mise en relation entre des configurations de signes (réduites ou augmentées). Le tout servant de « preuve ». Deuxièmement, la signification des signes n'est pas une propriété prise en compte par la « machine ». Cependant, ces signes peuvent, eux, être interprétés ou sont interprétables par des humains. Mais cette interprétation est extérieure à la machine et elle est non pertinente pour son fonctionnement.

Une formulation moins « physique », mais des plus importantes, se retrouvera chez Hobbes ([1651], 1962). Celui-ci exprimera par sa fameuse formule « *cogitare est computare* » que raisonner, c'est-à-dire penser, est une computation ; computation qui consiste en un ensemble d'opérations appliquées sur des *symboles* nominaux généraux (*nomina*) signifiant la pensée comme celles qu'on effectue sur des opérations de soustraction et addition c'est-à-dire : « *(reasoning) is nothing but reckoning (that is, adding and subtracting) of the consequences of general names agreed upon, for the marking and signifying of our thoughts* » (Hobbes, 1651).

Leibniz (1646-1716) reformulera cette idée en termes plus synthétiques de *calculus ratiocinator*. Il précisera une propriété spécifique d'un tel calcul : il doit être une *combinatoire* de symboles, c'est-à-dire un ensemble d*'opérations sur des symboles*. Ce calcul forme ainsi une *lingua caracteristica*. Idée que Condillac (1714-1780) reprendra à son tour en affirmant que si cette langue est parfaite, elle est une *algèbre* : « l'algèbre est une langue bien faite et c'est la seule ! »

Ce sera Boole (1847) qui ajoutera la fibre la plus importante au lien entre calcul et computation. Pour lui, un calcul sur symboles repose, certes, sur la vérité des lois de combinaisons des symboles, mais surtout il ne doit pas dépendre de la *signification* de ces *symboles* : « *Their validity does not depend on the significance of the symbols which they involve, but only on the truth of the laws of their combination*. » (Boole, 1847 : 2, 7)

Selon Kneale (1948), ce serait là l'une des importantes « découvertes » de Boole : « *there could be an algebra of entities which were not numbers in any ordinary "sense" and that the "numeric" laws governing these entities need not be arithmetic* » (Kneale, 1948 : 160).

Ainsi, dans cette tradition, il existe un lien serré entre raisonner, computer, calculer, combiner et manipuler des signes. Autrement dit, un calcul valide est une combinatoire appliquée à des signes, et il est effectué sans égard à la signification de ces signes.

Une telle thèse est importante. En effet, il ne suffit pas pour avoir de la computation au sens de Hobbes, mais surtout de Boole, qu'une machine physique configure et reconfigure aléatoirement des signes. Il faut que ces configurations soient des combinaisons régies par les lois et ceci hors de leur signification.

Aujourd'hui, cette thèse sera exprimée en termes linguistiques : *un calcul ou une computation est une opération qui, appliquée de manière purement syntaxique à des ensembles signes, produit d'autres ensembles de signes et ceci sans égard à leur sémantique*.

### *5.1 Les modèles computationnels et la validité*

Cette dernière thèse demeure générale. Elle ne précise pas encore les conditions nécessaires et suffisantes pour qu'un calcul ou une combinatoire de signes (sans signification) soit *valide*. En effet, on pourrait facilement créer une machine qui manipule de manière réglée des configurations de signes sans égard à leur signification. Mais il resterait à savoir si ces manipulations produisent des formules valides. Autrement dit, comment décider si ces formules sont valides ? Ce qui nous ramène à notre question : qu'est-ce qu'un calcul valide sur des signes ?

La réponse à cette question est connue ; elle se formule ainsi : un calcul est valide s'il est possible de déterminer ou de décider si une formule appartient ou n'appartient pas au système symbolique. Autrement dit, il faut montrer qu'il existe des conditions nécessaires et suffisantes, c'est-à-dire : des procédures fermes de *décision ou de calcul* de la validité d'une formule[7].

En 1900, Hilbert reprendra cette question sous diverses formes dans les fameux problèmes qu'il avait soumis aux mathématiciens. On peut les paraphraser ainsi : Sous quelles conditions une formule d'un système symbolique de type arithmétique peut-elle être dite décidable ou calculable ? Plusieurs mathématiciens tenteront de répondre à cette question. Une première solution est proposée par Church (1936) : une formule mathématique sera calculable si elle est récursive. Mais c'est Turing qui apportera la solution qui sera ultérieurement la plus acceptée : un calcul est valide s'il est le résultat de l'application de *procédures effectives* sur les symboles, c'est-à-dire s'il est le résultat des opérations qu'une machine physique (appelée plus tard Machine de Turing) pouvait appliquer à des symboles déposés sur un support physique (— papier). On dira alors que cette machine « computait » la solution. Plus tard, Turing démontrera que les manipulations que sa machine exécutait étaient équivalentes au calcul que Church avait proposé. Ce sera la thèse dite *Church-Turing*[8]. Par la suite, ce calcul computationnel sera aussi démontré équivalent à de nombreuses autres formes de calculs, par exemple : le calcul récursif de Kleene (1936), les règles de production de Post (1936), la logique combinatoire de Curry et Feys, (1958), les algorithmes de Markov (1960), les grammaires à états finis (Chomsky, 1957), les automates de von Neumann (1963) et bien d'autres.

En bref, si une formule d'un système symbolique est décidable, elle peut être traduite par un algorithme dont une forme d'écriture est un langage de programmation. Il y a donc un lien serré entre les systèmes symboliques de type mathématique et ceux de type computationnel. Si un système mathématique assure la calculabilité et la décidabilité de ces formules, alors il est aussi un système formel computable.

Cette thèse de la computation est bien exprimée autrement par la définition de Soare :

> A computation is a process whereby we proceed from initially given objects, called inputs, according to a fixed set of rules, called a program, procedure, or algorithm, through a series of steps and arrive at the end of these steps with a final result, called the output. The algorithm, as a set of rules proceeding from inputs to output, must be precise and definite, with each successive step clearly determined. The concept of computability concerns those objects which may be specified in principle by computations, and includes relative computability (Soare, 1996 : 286).

Selon ces définitions, on peut donc voir un modèle computationnel comme une variante d'un modèle formel. En fait, il en est un sous-ensemble. Le modèle formel sera computationnel si les systèmes symboliques qu'il contient sont computables.

Enfin, Fodor et Pylyshyn (1988) proposeront une formulation plus sémiotique, linguistique et logique de ces thèses sur la nature de la computation. Un système est dit

[7] Si cette propriété peut être attribuée à toutes les formules du système, alors celui-ci est dit complet.

[8] La thèse Church-Turing établit une relation d'équivalence extensionnelle entre la formulation de Church et de Turing : « La même classe de fonctions partielles (et donc de fonctions totales) est obtenue dans chaque cas. » (Rogers, 1967 : 19)

computationnel si, dans le système symbolique, on peut identifier des constituants atomiques ou discrets, et si on peut définir des règles systématiques et productives (compositionnelles). Mais ils ajouteront une contrainte externe pour son usage effectif, c'est-à-dire il faut qu'on puisse associer une sémantique à toutes les formules d'un tel système symbolique.

Cette formulation sera reprise par les sciences cognitives et elle servira de thèse fondatrice pour les théories de l'esprit. Le cerveau deviendra la machine biologique de computation. C'est aussi par cette formulation que les courants de la sémiotique computationnelle se définiront. On verra ainsi certains systèmes sémiotiques, surtout ceux proches de la langue, comme pouvant être modélisés par des grammaires où la récursion est explicite. Dans ce cas, par la thèse de Turing-Church, on peut appliquer des modèles computationnels à ces systèmes symboliques ou sémiotiques.

Malgré la rigueur et la richesse de ce modèle computationnel, il sera difficilement accepté par les sémioticiens. Il impose d'importantes contraintes à l'analyse. On ne voit pas que de tels modèles soient applicables à des tâches d'analyse sémiotique comme, par exemple : *l'identification d'un style musical, la chorégraphie d'une danse, la formation des concepts, l'évolution de thèmes en littérature, les structures narratives, les liaisons intertextuelles, les biais épistémiques, les tendances architecturales, les conditions de production des discours, les structures argumentatives, la production d'une toile*, *la préparation d'une exposition, la transmission des mêmes,* etc.

Il est difficile de voir comment ces objets ou situations sémiotiques peuvent recevoir une modélisation computationnelle qui respecte la récursivité, l'algorithmie, la grammaticalité, etc. C'est fort probablement en raison de ces difficultés que plusieurs sémioticiens refusent d'en appeler à des modélisations computationnelles pour décrire et expliquer des réalités sémiotiques. Mais ceci n'est pas la seule raison. Il existe un problème encore plus profond : celui de la non-computationnalité. Et on ne peut l'esquiver.

### *5.2 La non-computation*

Liée à la thèse de la computationnalité des systèmes symboliques est la question suivante : se pourrait-il que certaines formules d'un modèle mathématique ne soient pas calculables et donc ne soient pas computables ?

Ce fut là un problème qui, sur le plan historique, fut souvent discuté en mathématique, car l'idéal d'une langue rationnelle ou combinatoire qu'avait proposé Leibniz fut régulièrement remise en question par plusieurs mathématiciens. Nous ne soulignerons ici que quelques points qui nous semblent pertinents relativement à la manipulation des symboles. Ils illustrent les problèmes intrinsèques de la calculabilité et ouvrent la porte à la non-computationnalité.

#### *Cas 1 : Les symboles d'abréviation*

Le premier cas est celui des symboles d'abréviation. Il interrogea sérieusement la calculabilité. Classiquement, on trouve dans les langues mathématiques des chiffres, des lettres et des opérateurs ($+$, $-$, $=$, etc.) et même des positions (colonnes, espaces, surélèvement, etc.). Les règles de leur manipulation furent toujours connues des mathématiciens. Mais le développement des mathématiques introduisit d'autres types de symboles qui devinrent problématiques pour le calcul. Parmi ceux-ci se trouvent les symboles d'abréviation.

Ainsi, dans sa *Lingua universalis,* Leibniz introduisit un de ces types de symboles pour représenter la multiplication d'un chiffre par lui-même. Cette opération s'écrivait habituellement $7 \times 7$. L'écriture de cette formule fut changée par l'insertion d'un symbole complexe constitué d'un chiffre plus un surélèvement, comme dans $7^2$ et que l'on comprend aujourd'hui comme la mise au carré ou la mise en puissance 2. Ce symbole en apparence simple est en fait une abréviation qui, lorsque le chiffre est surélevé, exprime un nombre d'itérations de la multiplication de la variable ou d'un chiffre par lui-même. Par exemple ; $y^4$ est l'abréviation de $y \times y \times y \times y$. Une telle abréviation est évidemment très utile et permet des écritures syncopées pour des puissances fortes, par exemple : $7^{20}$.

Leibniz proposa aussi des symboles d'abréviation pour plusieurs autres opérations. Par exemple : un grand « S » majuscule appelé peu après par Bernoulli (1654-1705) le symbole de l'intégrale pour remplacer la sommation effectuée dans l'intégration de la surface sous la courbe. Euler, en 1755, remplacera les sommations généralisées par un symbole spécifique : grand sigma « $\sum$ ». Certains mathématiciens allèrent encore plus loin dans cette création de symboles d'abréviation. Ils en utilisèrent le symbole « δ » pour calcul de la dérivée, et le symbole « $\int$ » pour l'intégrale, etc. Ceux-ci semblaient opérer de manière analogue à des symboles permettant des manipulations sur des symboles de quantité. Cauchy donna une forme définitive à ce type de calcul utilisant ces abréviations. Les mathématiciens apprécièrent ces multiples symboles de types supérieurs. Ils permettaient des calculs sur les opérations elles-mêmes. On les appela : « calculs d'opérations ». Lagrange (1772) voyait une « nouvelle espèce de calcul » dans ces nouveaux types de représentations. Cette stratégie fut acceptée par plusieurs mathématiciens et elle est encore utilisée aujourd'hui.

Cependant, comme l'a bien montré Koppelman (1971) dans une étude historique très détaillée de mathématiciens britanniques, ceux-ci s'interrogèrent sur validité du calcul qui peut être effectué en utilisant ces formules abrégées. Woodhouse (1803) par exemple, a montré de manière très détaillée et habile que dans certains cas, la notation algébrique qui utilisait la présentation abrégée d'une équation donnait des résultats différents de la présentation non abrégée. Apparaitraient ainsi des paradoxes sinon des contradictions. Il donnait l'exemple de l'équation suivante, où deux divisions représentent sous forme abrégée des séries

$$\frac{1}{1+x} = \frac{1}{x+1}$$

Ces deux formules de division sont reliées par un signe =, ce qui affirme qu'elles sont équivalentes. Cependant, selon Woodhouse[9], si on les présentait effectivement par leur extension en série, on verrait qu'elles ne le sont pas. En effet, on obtiendrait pour la première l'équation la série suivante :

$$\frac{1}{1+x} = 1 + x + x^2 + x^3 + \cdots$$

Et pour la seconde, nous aurions la série suivante :

$$\frac{1}{x+1} = \frac{1}{x} + \frac{1}{x^2} + \frac{1}{x^3} + \cdots$$

On constaterait alors que ces deux séries ne sont pas équivalentes. Il s'ensuivrait alors que l'équation originelle, qui met en équivalence leur forme abrégée, est problématique. Woodhouse en concluait alors qu'on ne peut affirmer une équivalence entre leurs formes abrégées : « *with reference to their expansions it cannot be affirmed that* $\frac{1}{1+x} = \frac{1}{x+1}$. » (cité par Koppelman, 1971 : 276)

Ainsi, l'introduction de symboles d'abréviation, et ce n'est qu'un exemple parmi plusieurs autres, posa des problèmes au principe de calculabilité dans un système algébrique : la manipulation des symboles d'abréviation n'assurerait plus toujours la validité des calculs.

*Cas 2 : Les symboles de variables*

Le deuxième cas concerne les symboles de variables. La présence de ces symboles posera à son tour des problèmes importants de calculabilité. Car, comme l'ont montré Desclés et Cheong (2006), la manipulation de formules symboliques avec des variables n'est pas aussi simple que la manipulation de formules ne contenant que des constantes.

Dans le calcul algébrique classique, le symbole se présente comme une lettre de l'alphabet et il semble manipulable comme un chiffre. Par exemple : $7x$, $x + 4 = 7$.

Ce qu'il y a de particulier pour le symbole $x$, c'est qu'il introduit dans le calcul un symbole qui, au plan sémantique, désigne une valeur inconnue. De ce fait, sa manipulation met en jeu une problématique particulière quant à la productivité de la langue formelle qui l'utilise. Elle introduit dans les formules autre chose que la composition classique. En effet, ce type de symboles pour les variables exige, pour être bien manipulé, du moins dans certains cas, non pas la simple combinaison avec d'autres symboles, mais la *substitution* de certains symboles à celui de la variable et ceci afin que le calcul puisse être appliqué de manière valide. Par exemple, pour résoudre une équation comme la suivante : $y = \sqrt{-x}$, il faut

[9] Voir l'analyse détaillée dans Woodhouse (1803 : 12-15, 58).

substituer des chiffres. Cependant, comme on le sait, ce n'est pas n'importe quel domaine de référence dans lequel on peut puiser ces chiffres. Ainsi, dans la précédente équation, si $x > 0$, alors l'équation n'a pas de solution (à moins d'accepter l'ensemble des nombres imaginaires) alors que si $x < 0$, il y a une solution. Il en sera ainsi de plusieurs autres utilisations de la variable. En raison de ces problèmes cachés, Descartes fut un des premiers à douter de la pertinence du symbole de la variable qui entrainait dans certains cas des absences de solution. Carnot les critiquera aussi en raison des paradoxes qu'elles pouvaient générer.

En fait, si on analyse plus attentivement le symbole de la variable, on constatera qu'il ne relève pas uniquement de la langue-objet. Il y a en lui quelque chose qui relève du métalangage. En effet, pour manipuler correctement le symbole de la variable, il faut d'une part savoir quel symbole est effectivement celui qui peut être manipulé comme un symbole de chiffre, et d'autre part, il faut que soit défini le domaine des valeurs qui pourront être à substituer à cette variable. Autrement dit, sous le symbole de la variable se cache en ensemble d'opérations que l'on résume sous le concept de substitution. Mais un tel ensemble d'opérations cachées introduit des difficultés importantes dans le calcul. Car les variables impliquent implicitement ou explicitement un croisement de deux énoncés distincts, l'un appartenant au langage-objet et l'autre au métalangage. Le premier indique quel symbole est la variable, le second en appelle à la sémantique du symbole. Ce qui affecte évidemment le principe même de la calculabilité des formules qui justement reposaient sur l'absence d'une sémantique.

Pour éviter ces problèmes, Church (1936) proposera une solution au calcul des fonctions algébriques avec variables en introduisant la notation lambda. Celle-ci permettait au sein même des formules de préciser les valeurs admissibles pour les variables. Par exemple, l'expression $(\lambda x)(x^2 = 4)$ déterminera sans ambiguïté les valeurs spécifiques que peuvent prendre les expressions fonctionnelles, c'est-à-dire elle déterminera quelle valeur de la variable peut rendre cette fonction vraie ou fausse. Sans pour autant introduire dans la formulation les expressions vraies ou fausses, Curry et Feys (1958) proposèrent une solution radicale : une langue formelle sans variable : la logique combinatoire.

Ces deux exemples illustrent la complexité de cette question de la calculabilité dans les systèmes formels. Elle est toujours l'objet de nombreux débats entre mathématiciens[10]. Comme nous le disions plus haut, en posant sa question de la calculabilité en 1900, Hilbert[11] mettait en évidence que certaines équations connues (par exemple, $x + 8 = 5y,\ x^2 = 2y^2$ – équations dites diophantiennes) étaient sans solution. Le lecteur attentif remarquera que les

[10] La découverte de divers paradoxes, par exemple ceux internes à la théorie de l'actualité de l'infini chez Cantor, ont déclenché de sérieux doutes sur les fondements de cette calculabilité. Et les solutions logiques de Frege n'avaient pas réussi à convaincre les mathématiciens. Elles furent vite confrontées à des paradoxes mis en évidence par Russell et Whitehead. Ce qui apparaissait fonder un calcul se voyait soudainement confronté à de la contradiction interne.

[11] Il est le grand problème que Hilbert avait proposé en 1900 lors de sa conférence au *Congrès international des mathématiciens* de Paris. Il faut remarquer cependant qu'elle était déjà débattue par Dedekind.

deux équations précédentes sont syntaxiquement bien faites. Pourtant, elles ne possèdent pas de procédure connue de décision pour trouver la solution.

Par son théorème d'incomplétude, Gödel (1931) démontra que, pour tout système symbolique axiomatique, il était possible, malgré la manipulation syntaxique formelle de symboles, de générer aussi bien une configuration P de symboles que son contraire non-P. Une telle position remettait radicalement en question la possibilité d'un calcul valide pour les langages formels. Turing à son tour démontra quelque chose de similaire relativement aux fonctions que sa machine ne pouvait pas computer. En 1939, il proposa une solution originale pour rendre computables des fonctions arithmétiques non computables : il en appelait à des « oracles » : *« With the help of the oracle we could form a new kind of machine (call them o-machines), having as one of its fundamental processes that of solving a given number-theoretic problem* » (Turing, 1939 : 161).

Ultérieurement, on découvrira que plusieurs fonctions mathématiques, bien que correctement formées, posent des problèmes majeurs de calcul. De fait, en mathématique, l'existence de ces types de fonctions non computables n'est pas un fait rare. Effectivement, les fonctions computables ne sont qu'un sous-ensemble très restreint des fonctions mathématiques.

Cette question de la non-computationnalité est encore plus problématique qu'on le pense[12]. En effet, il existe des systèmes formels qui, bien qu'utilisant des fonctions computables, peuvent devenir si complexes qu'ils se transforment en systèmes non computables. Chaitin (1998) a récemment montré que plus les systèmes symboliques sont complexes, plus ils incluront de la redondance (ce qui assure une stabilité au système). Et en conséquence, plus cette redondance peut entraîner la non-computationnalité du système formel.

Voilà autant de contraintes qui, dans de nombreuses recherches scientifiques, entrainent des problèmes de précision, de prédiction, de certitude, de validité, de complétude, etc. et donc, de non-calculabilité, : « *Undecidability and incompleteness are everywhere, from mathematics to computer science, to physics, to mathematically-formulated portions of chemistry, biology, ecology, and economics.* » (Chaitin, Doria et da Costa, 2012 : 2)

Et pour certains informaticiens contemporains, les fonctions non computables sont théoriquement beaucoup plus intéressantes que les fonctions computables : « *The subject (of computation) is primarily about incomputable objects not computable ones* » (Soare, 2009 : 59).

Bref, les systèmes mathématiques computationnels sont un ensemble éminemment restreint des systèmes mathématiques. La grande majorité des fonctions mathématiques sont non computables[13].Cette thèse de la non-computationnalité sera assurément la bienvenue par les sémioticiens pour les artefacts sémiotiques ne peuvent pas tous être modélisés de manière

[12] On retrouve des problèmes analogues dans les logiques dites paraconsistantes ; voir Chaitin, Doria et da Costa (2012).

[13] Comme l'ont montré plusieurs mathématiciens et informaticiens, plus particulièrement Chaitin (1998), la non-décidabilité est partout dans les systèmes formels.

computationnelle. Mais pour être encore plus radical, nous dirons que, comme en mathématique, en raison de la complexité des artefacts sémiotiques, il est infiniment plus probable qu'ils ne soient pas modélisables de manière computationnelle. Bref, le domaine de la sémiotique serait-il avant tout de type non-computable ?

Comme on le voit, cette notion de non-comptabilité vient déstabiliser la construction des modèles formels et des modèles computationnels. Car même si un modèle formel était possible, il n'est pas assuré que les fonctions découvertes, générées ou construites seraient calculables et donc computables ! Autrement dit, tout modèle formel n'est pas par définition un modèle computable.

Cette difficulté de la notion de computationnalité nous permet de préciser les contraintes inhérentes à un lien possible entre sémiotique et computation. Une théorie sémiotique de type computationnelle ne peut exister que si elle en appelle à de la modélisation formelle de type mathématique (non pas nécessairement quantitatif), dont les énoncés, formules ou équations permettent la calculabilité. Ce n'est qu'à cette condition qu'une sémiotique peut être computationnelle.

En sémiotique, comme Peirce l'avait vu[14], ceci signifie que les systèmes symboliques doivent être contrôlés et qu'en conséquence, ces systèmes rencontreront leurs propres limites. Une machine, selon Peirce, opère toujours à partir de paramètres et de règles qui ne sont pas les siennes, mais qui lui sont données de l'extérieur.

On voit que le lien possible entre la sémiotique et la computation est situé en un endroit très spécifique : *Si une analyse sémiotique peut inclure dans sa description et son explication d'une réalité sémiotique un modèle formel possédant des opérations de manipulations valides, c'est-à-dire y inclure un calcul valide sur les symboles spécifiques, alors ce modèle sera équivalent à un modèle computabl*e.

En contrepartie, il ne sera pas possible de construire un modèle computationnel pour des réalités sémiotiques qui ne posséderaient pas toutes les conditions nécessaires pour les modéliser de manière computationnelle, algorithmique, récursive, combinatoire, etc. Postuler que la modélisation computationnelle pour traduire adéquatement tant la multiplicité et la complexité des réalités sémiotiques est une hypothèse que peu de chercheurs seraient prêts à soutenir.

## 6. Le modèle informatique

Le troisième modèle est de type informatique. Tout comme les autres, il assiste la description et l'explication d'une réalité, mais cette fois, il effectue cette tâche en faisant

[14] Il faut se rappeler que Peirce a travaillé sur la construction de machines logiques. Son père avait tenté d'installer une machine de Babbage à l'observatoire d'Albany. Et pendant son séjour à Johns Hopkins, il avait travaillé sur une machine logique avec Allan Marquand, l'un de ses étudiants. Il demeure cependant critique quant à la possibilité de raisonnement pour cette machine, c'est-à-dire de sa capacité de résoudre certains problèmes logiques et mathématiques. Voir Peirce (1976b : 625-632) et da Silveira (1993).

appel à une technologie. Il est intimement lié au modèle formel et surtout au modèle computationnel. À première vue, on aurait tendance à identifier la notion de computation avec celle d'ordinateur. En effet, dans le langage populaire, les deux sont souvent prises comme synonyme. On dira facilement que la propriété d'un ordinateur est la computation parce qu'il peut calculer une fonction. Mais cette formulation est ambiguë. Or, il y a une différence entre la notion de computation et la notion d'ordinateur.

La notion de computation, avons-nous dit, appartient à une théorie mathématique. Elle touche la question de la calculabilité des fonctions mathématiques. Ce qui est computable formellement est une *fonction* au sens strict du terme. La notion d'ordinateur, pour sa part, appartient au domaine d'une technologie physique qui permet d'effectuer cette computation. Et ce n'est que si une fonction est computable qu'un ordinateur peut la computer. Qui plus est, pour une même fonction calculable, il y a plusieurs machines physiques qui peuvent la calculer. La machine de Turing n'en qu'une parmi d'autres. Si la fonction n'est pas trop complexe, même des humains peuvent concrètement la calculer. Avant Turing d'ailleurs, des humains appelés calculateurs ou *computor* faisaient ce travail. Bref, la computationnalité est une notion logiquement antérieure à celle d'ordinateur. En conséquence, un modèle computationnel est différent d'un modèle informatique[15].

Ainsi, ce qui caractérisera un modèle informatique est la forme d'architecture qui sera donnée à l'ordinateur. Et en cela, tous les modelés informatiques ne seront pas identiques. Cette architecture peut varier et rendre plus ou moins efficaces, et même plus ou moins possibles, les opérations de calcul. Tout dépendra de la complexité des fonctions à calculer et des composantes contenues dans la technologie informatique.

Par exemple, l'architecture de la petite machine de Turing, avec ses roues, son bras et ses rouleaux de papier (voir la figure 2), peut « théoriquement » tout calculer, mais la vitesse et le nombre d'opérations peuvent être si grands qu'il devient impossible d'effectuer le calcul d'une fonction complexe en un temps fini.

### *La machine de Turing*

[15] Il faut nuancer cette formulation ici. Le fait pour une fonction d'être réalisée par une machine physique comme la machine de Turing ou sa variante, une machine de von Neumann, ou son équivalent, un ordinateur, garantit que la fonction est computable. Mais il n'en demeure pas moins que sur un plan intensionnel, l'ordinateur et la machine de Turing ne sont pas identiques à une fonction computationnelle. Ils lui sont simplement extensionnellement équivalents. Car il y plusieurs autres types de machines physiques qui peuvent effectuer une computation.

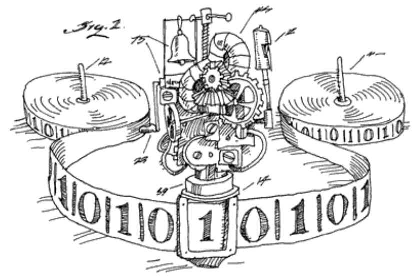


Figure 2

Vers les années 1940, von Neumann proposa une architecture plus sophistiquée pour les ordinateurs. Dans ses formes primitives, cette architecture était composée d'une mémoire, d'unités arithmétiques et logiques, d'unités de contrôle et des dispositifs d'entrée et de sortie (voir la figure 3).

*Schématisation de l'architecture de von Neumann*

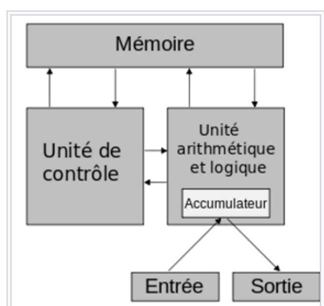


Figure 3

Malgré l'amélioration de la puissance et du temps apportée par cette architecture, en raison de leur complexité intrinsèque, toutes les fonctions formellement computables ne peuvent pas toujours être traitées concrètement.

Par exemple, un ordinateur à l'architecture classique von Neumann peut computer des équations de la mécanique newtonienne (comme la loi de la gravité), mais en raison du temps de traitement nécessaire, il peut être impossible de traiter concrètement certaines fonctions mathématiques d'un modèle formel chaotique pour décrire et expliquer la dynamique gravitationnelle d'un typhon.

Évidemment, cette architecture classique de base de von Neumann a grandement évolué depuis ses origines. Une multiplicité et une diversité de types de composantes se sont ajoutées, tels des compilateurs, des traitements parallèles des langages de programmation, des périphériques d'entrée et de sorties, des bases de données, les codes mobiles, etc.

Aujourd'hui, l'architecture des ordinateurs est encore plus complexe. La miniaturisation des composantes, mais surtout les ordinateurs distribués, la grille informatique (*grid computing*), le nuagique, les serveurs centralisés, etc. permettent une vitesse et une puissance de traitements computationnels impossible auparavant (voir la figure 4).

*Un réseau d'ordinateurs et de serveurs collaborant à une computation effective*

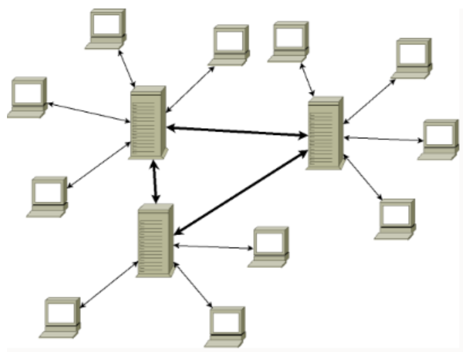

Figure 4

Mais ces architectures ont un ajout majeur : elles ont la capacité de traiter des fonctions computationnelles d'une grande complexité en un temps acceptable pour des recherches scientifiques. On peut ainsi cumuler, fouiller, comparer, archiver, visualiser, transférer des données massives. Elles peuvent traiter d'immenses catalogues de règles, des grammaires formelles complexes, de grands calculs bayésiens, des champs de Markov sophistiqués, des systèmes dynamiques chaotiques, de l'apprentissage machine, de l'apprentissage profond et de l'intelligence artificielle, etc.

En raison de la puissance de ces architectures, certaines sciences ont ainsi pu construire de puissants modèles informatiques pour explorer plus en profondeur leur territoire de recherche. Pensons à la génétique, la climatologie, l'astrophysique, les réseaux sociaux sémantiques, la gestion distribuée des entreprises, la médecine, l'environnement, la cognition, la robotique, la biologie informatique, etc.

Ces puissantes architectures ne sont pas sans modifier et changer profondément la forme et les conditions mêmes de la pratique scientifique. En effet, bien que la science puisse construire des modèles formels et computationnels d'une belle richesse et d'une grande complexité, ceux-ci seront inefficaces pour décrire et expliquer des phénomènes s'il n'existe pas une architecture informatique capable de computer concrètement les fonctions que ces modèles contiennent. En ce sens, l'ordinateur contemporain a ouvert de nouvelles possibilités à la science.

Comme l'ont constaté récemment certains épistémologues (Thagard, 1999 ; Giere, 1988, 1999 ; Latour, 1986 ; Vallverdu i Segura, 2009), l'ordinateur joue un rôle de plus en plus important dans la pratique et la théorisation scientifique. Par exemple, il renouvelle les formes expérimentales, consolide les démonstrations, les preuves, les validations, les falsifications, les évaluations, la dynamique d'échange, de contrôle, de publication, etc. Bref, il modifie le raisonnement scientifique lui-même. Il n'est plus simplement nomologique-déductif. La théorie opère avec et entre des modèles. Et les modèles informatiques en sont une forme importante et essentielle dans la pratique scientifique contemporaine.

Il faut cependant se rappeler que cette puissance informatique ne peut tout faire. Car, rappelons-le, un ordinateur ne peut « calculer » que ce qui est computable. Et ce qui est computable est une infime partie de ce qui est formellement modélisable. De plus, sur le plan de sa structure même, cette technologie n'est pas infaillible. L'erreur peut s'y introduire à plusieurs endroits tant dans les données que dans le code, les programmes, la coordination, les communications, et même l'infrastructure physique elle-même. On se rappellera le problème physique de la puce Pentium 1994[16], qui ramenait toujours les mêmes résultats pour certaines divisions en raison d'une coupure trop rapide de la virgule flottante. Ceci affectait les calculs complexes, surtout ceux fondés sur des modèles chaotiques sensibles aux conditions initiales.

La sémiotique n'échappera pas à cet impact de l'ordinateur. Plus il existera de modèles formels computables d'artefacts sémiotiques, plus la sémiotique utilisera elle aussi des modèles informatiques. De fait, plusieurs disciplines des sciences humaines effectuent de la recherche sur des artefacts sémiotiques et plusieurs en appellent à des modèles sémiotiques qui, de plus en plus, introduisent dans leur démarche des modèles formels computationnels. L'utilisation de l'ordinateur y est de plus en plus présente. Pensons aux études des représentations sociales en sociologie, des réseaux sociosémantiques en communication, des analyses conceptuelles en philosophie, des courants littéraires en lettres, des fouilles archéologiques, des archives historiques, des cultures en anthropologie, des discours politiques en sciences politiques, de la jurisprudence en droit. Sans parler des arts et des médias. Qui plus est, et de manière très forte, les humanités dites numériques se définissent par un rapport à la modélisation informatique (Katz, 2005 ; Schreibman et coll., 2008 ; McCarthy, 2008 ; Unsworth, 2002).

Bien que ces artefacts soient complexes, et que certaines de leurs composantes ne soient pas modélisables formellement et computationnellement, certaines composantes importantes de la démarche sémiotique peuvent, elles, être modélisées computationnellement et ensuite soumises à un traitement informatique. C'est précisément dans ces interstices que les architectures informatiques contemporaines, avec leurs immenses reposoirs et leur puissance de traitement algorithmique, ouvrent de nouvelles voies de recherche à la sémiotique.

Prenons par exemple les puissants reposoirs de données. Ces données massives ne sont pas sans modifier les « observables » sémiotiques et imposer de nouvelles formes

[16] https://en.wikipedia.org/wiki/Pentium_FDIV_bug.

d'analyse. Si autrefois on pouvait privilégier quelques artefacts sémiotiques, par exemple une dizaine de récits, de mythes, un ou deux textes d'un auteur, aujourd'hui, les chercheurs des sciences humaines sont soudainement confrontés à des données massives, issues de projets gargantuesques de numérisation de patrimoines textuels, iconiques, sonores médiatiques, etc. Pensons aux Google Books, Facebook, Twitter, Wikipédia, GLAM wiki, aux musées virtuels, etc. Pensons à ces reposoirs qui contiennent de plus en plus de données sémiotiques multimodales où se rencontre une diversité de types de signes (texte, image sons, etc) (voir Nowakawska, 1981).

On peut être pour ou contre ces données massives en raison de leur non-pertinence, non-équivalence, ambiguïté, superficialité, etc. (voir la critique de Boyd, 2011,) ils n'en recèlent pas moins de questions sémiotiques importantes qui imposent un recours à un traitement computationnel même minimal.

Outre ces reposoirs, les architectures informatiques contemporaines permettent d'annoter les données représentant les artefacts sémiotiques sous forme de métadonnées. Certaines de ces métadonnées peuvent être ajoutées automatiquement. Mais comme les annotations sont toujours des actes interprétatifs, elles ne peuvent pas toujours être effectuées de manière algorithmique ; aussi faut-il plutôt assister leur insertion de manière manuelle. Dans les deux cas, on trouve des modules informatiques de plus en plus riches qui permettent de l'annotation éditoriale, lexicale, syntaxique, sémantique, pragmatique, sociale, culturelle, médiatique, etc. S'ajoutent aussi à ces modules des outils interactifs qui permettent d'adapter, de modifier, d'enrichir les métadonnées. Ainsi, des explorations sémiotiques classiques peuvent être enrichies. Pensons aux analyses de discours, de sentiments, d'arguments, de narrativité, de concepts, de thèmes, de réseaux sociaux, etc.

Les architectures informatiques contemporaines permettent aussi d'explorer des modèles formels computationnels de grande complexité, par exemple : les calculs bayésiens, les systèmes dynamiques, chaotiques, les réseaux de neurones, l'algèbre vectorielle, les classifieurs, l'apprentissage machine, l'apprentissage profond, les grammaires catégorielles, les réseaux sémantiques, etc. Lorsqu'appliquées à des réalités sémiotiques, des propriétés difficilement identifiables « à la main » peuvent être explorées pour faire émerger des hypothèses d'interprétation. Ainsi sont explorées des fouilles d'invariants cachés, des régularités sémiotiques, assurées ou probables, récurrentes et interreliées, etc. De même, de nouveaux domaines sémiotiques apparaissent : la sémiométrie (Lebart et coll., 2003), la lecture distante (Morreti, 2005), le forage d'arguments (Lippi et Torroni, 2016) la lecture et l'analyse conceptuelle assistée par ordinateur (LACTAO) (Meunier, 2009).

Outre ces reposoirs, ces métadonnées, ces analyseurs, on trouve de nouvelles formes d'interfaces 2D, 3D de visualisation. Celles-ci permettent d'effectuer des parcours sémiotiques dynamiques et des inférences nouvelles, des critiques originales (Greengrass et coll., 2008).

Enfin, l'interconnectivité des ordinateurs entraine de nouvelles pratiques de recherche telles les recherches collectives et partagées (par exemple, HASTAC, voir Davidson et coll., 2004). Des accès libres aux publications, des logiciels libres, des centres de recherches des nouvelles formes d'évaluation. Sans parler de la formation des étudiants (Mactavish et Rockwell, 2006).

Bref, les nouvelles architectures informatiques invitent à un changement dans la recherche sémiotique (Hale, 2017). Sous une forme ou une autre, elles rendent possible une sémiotique computationnelle qui, cependant, doit être comprise, non pas comme une sémiotique automatique, mais qui plutôt une assistance à la pratique classique de la recherche sémiotique : une sémiotique assistée par ordinateur.

## Conclusion : le défi du lien computation et sémiotique

Dans cette recherche, nous avons visé à préciser le lien technique entre la sémiotique et la computation. Au début de cet article, nous avons dessiné brièvement les deux grands courants de recherche qui manifestent explicitement une présence de la sémiotique tant de la modélisation formelle, de la modélisation computationnelle que de la modélisation informatique. Mais force est de constater qu'aucun des deux courants, sauf exception, n'a pris son envol. Il y a peu de chances que de l'intérieur de leur discipline, les informaticiens reconnaissent leur démarche comme incluant de la sémiotique. Les sémioticiens continueront à prendre la voie pragmatique et herméneutique. Quelques structuralistes et formalistes peut-être ici et là exploreront une modélisation formelle, computationnelle et informatique.

Mais les véritables raisons sont plus profondes. Dans son ensemble, le domaine de recherche de la sémiotique ne pourra pas facilement intégrer une modélisation formelle computationnelle et informatique. La raison principale, rappelée constamment par les sémioticiens, est que la recherche sémiotique est essentiellement construction de théories qui portent sur l'interprétation des signes qui constituent un artefact sémiotique.

Ceci limite grandement la possibilité d'un lien entre sémiotique et computation, et ultimement, l'utilisation de l'ordinateur. En effet, bien qu'une technologie créé l'illusion, souvent spectaculaire d'ailleurs, de manipuler du sens et de la signification, elle ne peut, par définition que manipuler des signes sans signification. À l'entrée, un ordinateur ne reçoit que des symboles et il ne peut que les reconfigurer en d'autres symboles. Comme le souligneront plusieurs chercheurs, dont, de manière récurrente, Searle (1980) et Harnad (1990), pour le domaine cognitif, un tel le traitement qu'effectue un ordinateur est essentiellement de nature « syntaxique ». Une sémantique peut toujours être associée aux données, aux traitements et aux extrants, mais elle est toujours externe. Et c'est l'objectif même de la sémiotique de travailler au dévoilement de cette signification.

Plus profondément encore, comme nous l'avons souligné plus haut, il est fort possible que les artefacts sémiotiques qu'étudie la sémiotique ne présentent pas des propriétés et des structures que des modèles puissent formaliser de manière computationnelle. En conséquence, même un ordinateur ne pourra les traiter.

Malgré ces difficultés, le projet d'un lien entre sémiotique et computation demeure possible, mais ce lien sera limité et devra respecter la nature de la recherche sémiotique et la nature de la computation.

Pour établir ce lien, nous avons proposé de voir la pratique de la recherche sémiotique comme une démarche scientifique qui construit des interprétations. Or, la vision de pragmatique de la science voit justement ces interprétations comme des modèles. En ce sens, la sémiotique est, à sa manière, une science qui construit des modèles ; certains sont conceptuels, phénoménologiques, structuraux, logiques, herméneutiques, etc. Ces modèles relèvent des pratiques classiques de la sémiotique. Mais un lien avec la computation n'est possible que si et seulement si certains modèles sont formels, computationnels et informatiques. Ce sont ces trois types de modèles que nous avons tenté de présenter plus en détail dans cette recherche :

a) Un modèle *formel* identifie dans l'objet de recherche des relations entre des signes et vise à les exprimer dans une langue formelle. Certaines de ces relations peuvent être des relations de dépendances fonctionnelles.

b) Un modèle *computationnel* traduit ces relations fonctionnelles en fonctions computables et algorithmisables.

c) Un modèle *informatique* construit une architecture informatique qui permet de computer effectivement les modèles formels et computationnels.

Enfin, pour résumer la problématique du lien entre sémiotique et computation, il faut s'interroger sur la possibilité de respecter les conditions de construction de ces trois modèles, car il se pourrait que ce qui intéresse vraiment la sémiotique, ce soit des problèmes de grande complexité et dont les modèles formels sont constitués en majeure partie de fonctions non calculables ou non computables. Peut-être que, tout au plus, certaines fonctions pourraient être « approximées » tellement ces artefacts sont riches sur le plan sémiotique. Certes, une démarche scientifique modélisatrice serait bienvenue, mais aurait-elle vraiment réussi à approcher la richesse sémiotique de ces artefacts ? Et même si elle y touchait un tant soit peu, il est fort possible qu'elle rencontre le paradoxe de Chaitin de la complexité.

Pour notre part, devant ces difficultés intrinsèques de la computationnalité appliquée à la sémiotique, nous privilégions une solution pratique inspirée de la notion d'oracle de Turing. Ceux-ci peuvent être compris comme des modules *ad hoc*, mais computables et insérés dans un modèle computationnel. Ils interviennent uniquement à des moments particuliers dans la démarche, plus particulièrement quand les opérations sur les fonctions bloquent ou qu'elles sont non calculables intrinsèquement ou par leur complexité. Dans cette perspective pratique, les modules oracles computationnels accompagnent le sémioticien dans son interprétation propre mais ils ne substituent pas à lui. Ils sont ainsi des outils dans une chaine d'analyse. Dans certains cas, ils aideront à effectuer des opérations des plus simples comme, copier des notes d'analyse, mais dans d'autres cas plus sophistiqués, ils pourraient l'aider à détecter ou « approximer » une régularité au sein de données sémiotiques massives ou non qui l'intéressent.

Les recherches que nous explorons (Meunier et coll., 2003) sur la lecture et l'analyse des textes (LATAO) ont pris cette voie pratique. Une analyse de texte est une pratique classique en sémiotique. Mais au point de départ, nous savons qu'un texte est un artefact complexe qui possède une multitude de niveaux sémiotiques (Rastier, 1987). Il est fort probable que cette complexité rende difficile, voire impossible, la création de modèles formels, computationnels et un traitement informatique. Dans ce cas, ces modèles peuvent cependant inclure des sections, des modules oracles. Et le tout ne peut produire qu'une assistance à ce travail sémiotique sur des textes. Et la recherche consiste alors, pour une part, à découvrir quels sont les meilleurs assistants. Et il y a quelquefois des surprises merveilleuses, car Hermès dirige toujours la recherche.

**Notice biobibliographique**
Jean-Guy Meunier, PhD, est professeur associé au département de philosophie de l'UQAM. Il est rattaché au programme de doctorat en informatique cognitive et au programme doctoral de sémiologie. Il est directeur du Laboratoire d'analyse cognitive de l'information (LANCI). Ses recherches portent sur la sémiotique, l'analyse des textes assistés par ordinateur (LATAO) et les humanités numériques. Sur ces questions, il a publié plus d'une centaine d'articles. Il est membre titulaire de l'Académie internationale de philosophie des sciences (Bruxelles).

## Ouvrages cités

ANDERSEN, P. B. (1991) *A Theory of Computer Semiotics*. Cambridge, Cambridge University Press.

BACHELARD, R. (1979), « Quelques aspects historiques des notions de modèle et de justification des modèles », dans P. DELATTRE et M. THELLIER (éd.), *Élaboration et justification des modèles*, Paris, Maloine.

BACHIMONT, B. (1996), « Herméneutique matérielle et Artefacture : des machines qui pensent aux machines qui donnent à penser », thèse de doctorat de l'École Polytechnique.

BLEY, D. M., A. Y. ANDREW, M. I. JORDAN et J. LAFFERTY (2003), « Latent Dirichlet Allocation », *Journal of Machine Learning Research*, vol. 3, p. 993-1022.

BOLASKO, L. L. S. et A. SALEM (dir.), (1996), *Analisi Statistica dei Dati Testuali*, Rome, CISU.

BOOLE, G. (1847), *The Mathematical Analysis of Logic, Cambridge*, Macmillan, Barclay & Macmillan.

BOOLE, G. (1854), « Of Signs in General, and of the Signs Appropriate to the Science of Logic in Particular; also of the Laws to which that Class of Signs are Subject », dans *An*

*Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities*, New York, Macmillan (reprinted with corrections New York, Dover Publications, 1958).
BOYD, D. (2011), « Dear Voyeur, Meet Flâneur… Sincerely, Social Media », *Surveillance and Society*, vol. 8, nº 4, p. 505-507.
BRADLEY, J., (2008), « Pliny: A Model for Digital Support of Scholarship », *Journal of Digital Information*, vol. 9, nº 1, p. XX.
BRUNET, E. (1986), *Méthodes quantitatives et informatiques dans l'étude des textes*, Paris, Champion.
BUSA R. A. (2008), « Perspectives on the Digital Humanities », dans SCHREIBMAN, S., R. SIEMENS et J. UNSWORTH (dirs) (2008), *A Companion to Digital Humanities*, Malden, Wiley-Blackwell.
CANGUILHEM, G. (1977), *Idéologie et rationalité dans l'histoire des sciences de la vie*, Paris, Vrin.
CARMICHAEL, R. (1855), *A Treatise on the Calculus of Operations*, Londres, Longman, Brown, Green, and Longmans.
CARNAP, R. (1935), *Philosophy and Logical Syntax*, Londres, Kegan Paul, Trench, Trubner & Co.
CARNAP, R. (1937), *The Logical Syntax of Language*, Londres, Kegan Paul, Trench, Trubner & Co.
CARNAP, R. (1971), *Writings on the General Theory of Signs*, La Haye, Mouton.
CARTWRIGHT, N. (1983), *How the Laws of Physics Lie*, Oxford, Oxford University Press.
CHAITIN G. J. (1998), *The Limits of Mathematics*, Singapour, Springer.
CHAITIN G. J., F. A. DORIA, N. C. A. da COSTA, (2012), *Gödel's Way: Exploits Into an Undecidable World*, Boca Raton (Floride), CRC Press.
CHOMSKY, N. (1957), *Syntactic Structures*, La Haye, Mouton.
CHURCH, A. (1936), « An Unsolvable Problem of Elementary Number Theory », *The American Journal of Mathematics*, vol. 58, nº 2, p. 345-363.
CONDILLAC, É. B. T. de (1798), *Œuvres de Condillac. Essai sur l'origine des connoissances humaines*. Paris, Houel.
CURRY H. B. et R. FEYS, (1958), *Combinatory Logic*, vol. I, Amsterdam, North-Holland Publishing Company.
DACOS, M et P. MOUNIER (2014), *Les Humanités numériques : état des lieux et positionnement de la recherche française dans le contexte international*, Paris, Institut français.
DAVIDSON, C., BACSY, R. et J. SCHNAPP (2004), « The HASTAC Vision: Humanities, Arts, Science and Technology Advanced Collaboratory », *Visual Resources Association Bulletin*, vol. 30, nº 3, p. 43-50.
DE SOUSA, C. S. (2005), *Semiotic Engineering of Human-Computer Interaction*. Cambridge, MIT Press.

DESCLÉS, J.-P. (1990), « Archétypes cognitifs, transitivités, et intentionnalité », *Protée*, vol. 18, nº 2. p. 7-19.
DESCLÉS, J.-P.et K. S. CHEONG (2006), « Analyse critique de la notion de variable. Points de vue sémiotique et formel », *Mathématiques et sciences humaines / Mathematics and Social Sciences*, nº 173, p. XX.
DOUELHI, M. (2011), *Pour un humanisme numérique*, Paris, Seuil.
FEYERABEND P. ([1975], 2010), *Against Method: Outline of an Anarchistic Theory of Knowledge*, New York, Verso Books.
FLORIDI, L. (1999), *Philosophy and Computing*, Londres, Routledge.
FODOR, J. A. (1975), *The Language of Thought*, New York, Crowell.
FODOR, J. A. et Z. W. PYLYSHYN (1988), « Connectionism and Cognitive Architecture: A Critical Analysis », *Cognition*, vol. 28, nº 1-2. p. 3–71.
FRIGG, R. (2011), « Fiction in Science », dans J. WOODS (ed.), *Fiction and Models: New Essays*, Munich, Philosophia Verlag.
FRIGG, R. et S. HARTMANN (2012), « Models in Science », *The Stanford Encyclopedia of Philosophy*.
GIERE, R. N. (1988), *Explaining Science: A Cognitive Approach*, Chicago, University of Chicago Press.
GIERE, R. N. (1999), « Using Models to Represent Reality », dans L. MAGNANI, L. NERSESSIAN, P. THAGARD, (ed.), *Model-Based Reasoning in Scientific Discovery*, New York, Plenum Publisher, p. 41-57.
GÖDEL, K. (1931), « On Formally Undecidable Propositions of Principia Mathematica and Related Systems », dans J. van HEIJENOORT (dir.) *From Frege to Gidel: A Source Book in Mathematical Logic*, 1879-1931, Cambridge, Harvard University Press, p. 596-616.
GREENGRASS, M. et L. M. HUGHES (2008), *The Virtual Representation of the Past. Digital Research in the Arts and Humanities Series*, Londres, Routledge.
HABERMAS, J. (1973), *La technique et la science comme « idéologie »*, Paris, Gallimard.
HARNAD, S. (1990), « The Symbol Grounding Problem », *Physica D*, vol. 42, p. 335-346.
HARRIS, Z. (1968), *Mathematical Structure of Language*, New York, Interscience Publishers.
HAYLES, N. K. (2012), *How We Think: Digital Media and Contemporary Technogenesis*, Chicago, University of Chicago Press.
HALE, J. (2016), *Merleau-Ponty for Architects*, Abingdon, Routledge/Taylor and Francis.
HEMPEL, C. G. (1965), « Aspects of Scientific Explanation », dans C. G. Hempel (dir.), *Aspects of Scientific Explanation and Other Essays in the Philosophy of Science*, New York, Macmillan, p. 331-496.
HIIPPALA, T. (2013), « The Interface Between Rhetoric and Layout in Multimodal Artefacts », *Literary and Linguistic Computing*, vol. 28, nº 3, p. 461-471.
HILBERT, D. ([1899], 1971), *Les fondements de la géométrie*, Paris, Dunod.
HOBBES, T. ([1651], 1962), *Leviathan*, New York, Collier Books.
HOCKEY, S. M. (2000), *Electronic Texts in the Humanities: Principles and Practice*, Oxford, Oxford University Press.

HOFFMEYER, J. et D. FAVEREAU (2009), Biosemiotics. *An examination into the Signs of Life and the Life of Signs*, Scranton, University of Scranton Press.

HORKHEIMER, M. (1974), *Théorie traditionnelle et théorie critique*, Paris, Gallimard.

JACKENDOFF, R. (1987), *Consciousness and the Computational Mind*, Cambridge, MIT Press.

KATZ, S., N (2005), « Why Technology Matters: the Humanities in the Twenty-First Century », *Interdisciplinary Science Reviews*, vol. 30, nº 2, p. 105-118.

KLEENE, S.C. (1936), « General Recursive Functions of Natural Numbers », *Mathematische Annalen*, vol. 112, p. 727-742.

KNEALE, W. (1948), « Boole and the revival of logic », *Mind*, vol. 57, nº 226, p. 149–175.

KNEALE, W. et KNEALE, M. (1988), *The Development of Logic*, Oxford, Clarendon Press.

KNUUTTILA, T. (2011), « Modelling and representing: An artefactual approach to model-based representation », *Studies in History and Philosophy of Science*, vol. 42, p. 262-271.

KNUUTTILA, T. et A. LOETTGERS (2012), « Mathematization in Synthetic Biology: Analogies, Templates and Fiction », dans J. LENHARD et M. CARRIER (ed.), *Mathematics as a Tool, Tracing New Roles of Mathematics in the Sciences*, Amsterdam, Springer, p. 37-56.

KOPPELMAN, E. (1971), « The Calculus of Operation and the Rise of Abstract Algebra », *Archive for History of Exact Sciences*, vol. 8, nº 3, p. 155–242.

KUHN, T. S. (1962), *The Structure of Scientific Revolutions*, Chicago, University of Chicago Press.

LAGRANGE, J., (1772), « Sur une nouvelle espèce de calcul relatif à la différentiation et à l'intégration des quantités variables », *Œuvres*, vol. 3, p. 441- 476, Gallica, Bibliothèque nationale de France http://gallica.bnf.fr/ark:/12148/bpt6k229222d/f442.

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

LATOUR, B. (1986), « Visualization and Cognition: Thinking with Eyes and Hands », *Knowledge and Society*, vol. 6, p. 1-40.

LEBART, L. et A. SALEM (1988), *Analyse statistique des données textuelles*, Paris, Dunod.

LEBART, L., M. PIRON et J.-F. STEINER (2003), *La sémiométrie*, Paris, Dunod.

LEIBNIZ, G. W., (1969), *Philosophical Papers and Letters*, dans L.E. LOEMKER (dir.), Dordrecht, D. Reidel.

LEONELLI, S., (2007), « What is in a Model? Combining Theoretical and Material Models to Develop Intelligible, Modeling Biology: Structures, Behaviors, Evolution », dans M. LAUBICHLER, et G. B. MÜLLE (ed.), *Modeling Biology*, Cambridge, MIT Press.

LIEB, H. H. (1979), « Formal Semiotics vs Theoretical Semiotics », *Semiotic Unfolding*, vol. 1, Tasso Borbé (ed.), Berlin, Mouton Publishers, p. 175-180.

LIPPI, M. et P. TORRONI (2016), « Argumentation Mining: State of the Art and Emerging Trends », *ACM Transactions on Internet Technology*, vol. 16, nº 2, p. 10.

LIU, K. (2005), *Semiotics in Information Systems Engineering*, Cambridge, Cambridge University Press.

LULL, R., (1985), *Selected Works of Ramon Lull, 1232–1316*, Princeton, Princeton University Press.

MACTAVISH, A. et G. ROCKWELL (2006), « Multimedia Education in the Arts and Humanities », dans R. G. SIEMENS et D. MOORMAN, *Mind Technologies: Humanities Computing and the Canadian Academic Community*, Calgary, University of Calgary Press, p. 225-243.

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

MARTY, R. F (1984), « Normalisation et extension de la sémiotique de C. S. Peirce », *Semiotic Unfolding*, vol. 1, Tasso Borbé (dir.), Berlin, Mouton Publishers, p.185-193.

MARTY, R. F. (1990), *L'algèbre des signes. Essai de sémiotique scientifique d'après C. S. Peirce*, Amsterdam/Philadelphie, J. Benjamins Pub. Co.

MAYAFFRE, D. (2000), *Le poids des mots. Le discours de gauche et de droite dans l'entre-deux-guerres. Maurice Thorez, Léon Blum, Pierre-Etienne Flandin et André Tardieu (1928-1939),* Paris, Champion.

McKINNON, A. (1973), *Computational Analysis of Kierkegaard's Samlede Værker*, Leyde, E. J. Brill.

McCARTHY, W. (2008), « Modeling: A Study in Words and Meanings », dans SCHREIBMAN, S., R. SIEMENS et J. UNSWORTH (dir.) (2008), *A Companion to Digital Humanities*, Malden, Wiley-Blackwell, p. XX.

MEUNIER, J.-G (2012), « Humanités numériques, enjeux et méthodes », *Revue Texto*, vol. 17, nº 1 et 2, http://www. revue-texto. net/index.php? id=3028.

MEUNIER, J.-G. et D. FOREST (2009), « Lecture et analyse conceptuelle assistée par ordinateur : premières expériences », dans *L'annotation*, Paris, Hermès, p. 211-230.

MEYSTEL, A. et J. ALBUS (2002), « Intelligent Systems: Architecture, Design, and Control », dans *Wiley Series on Intelligent Systems*, Hoboken, J. Wiley and Sons, p. XX.

MORRETI, F. (2005), *Distant Reading*, New York, Verso Books.

MORRIS, C. W. (1964), *Signification and Significance. A Study of the Relations of Signs and Values*, Cambridge, MIT Press

MORRIS, C. W. (1971), *Writings on the General Theory of Signs*, La Haye, Mouton.

MORGAN, M. et M. MORRISON (ed.) (1999), *Models as Mediators, Perspectives on Natural and Social Science*, Cambridge, Cambridge University Press.

NADIN, M. (1977), « Sign and Fuzzy Automata », *Semiosis*, vol. 1, nº 5, p. XX.

NADIN, M. (2007), « Semiotic Machine », *The Public Journal of Semiotics I*, nº 1, p. 57-75.

NADIN, M. (2010) « Information and Semiotic Processes. The Semiotics of Computation », *Cybernetics and Human Knowing*, vol. 18, nº 1-2, p. 153-175.

NAGEL, E. (1961), *The Structure of Science: Problems in the Logic of Scientific Explanation*, New York, Harcourt, Brace & World.

NEUMANN, J. von (1945), *First Draft of a Report on the EDVAC*, https://sites.google.com/site/michaeldgodfrey/vonneumann/.

NEUMANN, J. von (1963), « The General and Logical Theory of Automata », dans A. H. TAUB (ed.), John von Neumann. *Collected Works*, Oxford, Pergamon Press, p. XX.

NEURATH, O. (1973), *Empiricism and Sociology*, Dordrecht, D. Reidel.
NEWELL, A. et H. SIMON (1976), « Symbol Manipulation », dans *Encyclopedia of computer science*, New York, Petrocelli/Charter, p. XX.
NEWELL, A. et H. SIMON (1976), « Computer Science as Empirical Inquiry: Symbols and Search », *Communications of the ACM*, vol. 19, nº 3, p. 113-126.
NOWAKAWSKA, M. (1981), « Formal Semiotics and Multidimensional Semiotic Systems », *Cybernetics and Systems. An International Journal*, vol. 12, nº 1-2, p. 83-102.
PAVEL, T. G. (1975), « Possible Worlds in Literary Semantics », *The Journal of Esthetics and Art Criticism*, vol. 34, nº 2, p. 165-176.
PEIRCE, C. S. (s.d.), « Miscellaneous Fragments [R]. MS [R] S104 », *Commens Digital Companion to C. S. Peirce*, http://www.commens.org/dictionary/term/semeiotic.
PEIRCE, C. S. (1931–1958), *Collected Papers*, Cambridge, Harvard University Press.
PEIRCE, C. S. (1976b), Caroline Eisele (ed.), *The New Elements of Mathematics*, vol. 4, La Haye, Mouton.
PELC, J. (1979), « On the Concept of Formal Semiotics », dans *Semiotic Unfolding*, vol. 1, Tasso Borbé (dir.), Berlin, Mouton Publishers, p. 269-276.
PETITOT, J. (1985), *Morphogenèse du sens : pour un schématisme de la structure*, Paris, PUF.
PETOFI, J. A. (1978), « A Formal Semiotic Text, Theory as an Integrated Theory of Natural Language (Methodological Remarks) », dans DRESSLER (dir.), *Current Trends in Textlinguistics*, Berlin/New York, W. de Gruyter, p. 35-46.
POST, E. L. (1944), « Recursively Enumerable Sets of Positive Integers and their Decision Problems », *Bulletin of the American Mathematical Society*, vol. 50, nº 5, p. 284-316.
POST, E. L. (1936), « Finite Combinatory Processes – Formulation », *Journal of Symbolic Logic*, vol. 1, nº 3, p. 103-105.
PYLYSHYN, Z. W. (1984), *Computation and Cognition: Toward a Foundation for Cognitive Science*, Cambridge, MIT Press.
RASTIER, F. (1987), *Sémantique interprétative*, Paris, PUF.
RASTIER, F. (2011), *Le grain et la mesure*, Paris, Champion.
REICHENBACH, H. (1938), *Experience and Prediction: An Analysis of the Foundations and the Structure of Knowledge*, Chicago, The University of Chicago Press.
RHEINBERGER, H. (1997), *Towards a History of Epistemic Things. Synthesizing Proteins in the Test Tube*, Stanford, Stanford University Press.
RIEGER, B. B. (1999), « Computing Fuzzy Semantic Granules from Natural Language Texts. A Computational Semiotic Approach to Understanding Word Meanings », dans M. H. HAMZA (ed.), *Artificial Intelligence and Soft Computing, Proceedings of the IASTED International Conference*, Anaheim/Calgary/Zürich (IASTED/Acta Press), p. 475-479.
ROCKWELL, G. (2003), « What is Text Analysis, Really? », *Literary and Linguistic Computing*, vol. 18, nº 2, p. 209-219.
ROGERS, H. (1967), *Theory of Recursive Functions and Effective Computability*, Cambridge, MIT Press

SALISBURY, J., (1929), *Ioannis Saresberiensis episcopi Carnotensis Metalogicon, Libri IV*, C. J. Webb (ed.), Oxford, Clarendon Press.
SAUSSURE, F. de ([1916], 1990), *Cours de linguistique générale*, Paris, Payot.
SCHLICK, M. (1979), *Philosophical Papers. Volume 1 (1909-1922)*, H. L. MULDER et B. F. VAN DE VELDE (dir.), Dordrecht, D. Reidel.
SCHREIBMAN, S., R. SIEMENS et J. UNSWORTH (dir.) (2008), *A Companion to Digital Humanities*, Malden, Wiley-Blackwell.
SEARLE, J. R. (1980), « Minds, Brains and Programs », *Behavioral and Brain Sciences*, vol. 3, p. 417-458.
SEBEOK, T. A. (1976), *Contributions to the Doctrine of Signs*, Bloomington, Indiana University Press.
SILVEIRA, L. F. B. da (1993), « Peirce e a contemporânea filosofia da ciência: uma dificil conversação », *Trans/form/ação*, vol. 16, p. 63-82.
SILVEIRA, L. F. B. da (1996) *Some Considerations about Semiotic Machines from the Point of View of Charles S. Peirce's Philosophy*, MS DEPT of philosophy, Sao Paulo, Sao Paulo University, http://www.inm.de/kip/semiotic/silveira_article.html.
SOARE, R. I. (2009), « Turing Oracle Machines, Online Computing, and Three Displacements in Computability Theory », *Annals of Pure and Applied Logic*, vol. 160, nº 3, p. 368-399.
SOARE, R. I. (1996), « Computability and Recursion », *Bulletin of Symbolic Logic*, vol. 2, nº 3, p. 284-321.
SOUZA, C. S. de (1993), « The Semiotic Engineering of User Interface Languages », *International Journal of Man-Machine Studies*, vol. 39, nº 5, p. 753-773.
SUPPES, P. (1967), « What is a Scientific Theory? », dans S. MORGENBESSER (dir.), *Philosophy of Science Today*, New York, Basic Books, p. 55-67.
TANAKA-ISHII, K. (2010), *Semiotics of Programming*, Cambridge, Cambridge University Press.
THAGARD, P. (1999), *How Scientists Explain Disease*, Princeton, Princeton University Press.
THAGARD, P. (2004), « Casual Inference in Legal Decision Making: Explanatory Coherence vs Bayesian Networks », *Applied Artificial Intelligence I*, vol. 18, p. 231-249.
THAGARD, P. (2012), *The Cognitive Science of Science: Explanation, Discovery, and Conceptual Change*, Cambridge, MIT Press.
TIJUS, C., J. BARCENILLA, B. C. LAVALETTE et J.-G. MEUNIER (2006), « The Design, Understanding and Usage of Pictograms », *Improving the production and understanding of written documents in the work place*, Amsterdam, Elsevier Publishers, p. 17-32.
TURING A. M. (1936), « On Computable Numbers, with an Application to the Entscheidungsproblem », *Proceedings of the London Mathematical Society*, Series 2, 42, p. 230-265.
TURING, A. M. (1939), « Systems of Logic Based on Ordinals », *Proceedings of the London Mathematical Society*, Series 2, 45, p. 161-228.

UNSWORTH, J. (2002), « What is Humanities Computing and What is it not? », *Jahrbuch für Computerphilologie*, vol. 4, p. 71-83, http://computerphilologie.tu-darmstadt.de/jg02/unsworth.html.

VALLVERDU I SEGURA, J. (2009), « Computational Epistemology and e-Science: A New Way of Thinking », *Minds & Machines*, vol. 19, p. 557-567.

Van DIJK, T. A. (1980), *Macrostructures*, Hillsdale, Lawrence Earlbaum Associates.

Van FRAASEN, B. C. (1970), « On the Extension of Beth's Semantics of Physical Theories », *Philosophy of Science*, vol. 37, nº 3, p. 325-339.

Van FRAASEN, B. C. (1989), *Laws and Symmetry*, New York, Clarendon Press.

Van FRAASEN, B. C. (2008), *Scientific Representation: Paradoxes of Perspectives*, Oxford, Oxford University Press.

VICKERS, P., J. FAITH et N. ROSSITER (2013), « Understanding Visualization: A Formal Approach Using Category Theory and Semiotics », *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, nº 6. p. 1048-1061.

WOODHOUSE, R. (1803), *The Principles of Analytical Calculation*, Cambridge, Cambridge University Press.