International Journal of Innovation and Applied Studies
ISSN 2028-9324 Vol. 21 No. 4 Nov. 2017, pp. 630-634

http://www.ijias.issr-journals.org/

Incomplétude et intelligence artificielle

A.-Roger LULA BABOLE

Département des Mathématiques et Informatique, Université de Kinshasa,
Faculté de Philosophie, RD Congo

ABSTRACT: The degree of freedom of a machine is organized and specify by a human been. Even it's well known that a machine is very quick and very reliable than a human been; this remain in the center because the algorithm was made by him. The calculable functions in TURING's machine are all calculable in intuitive point of view. So there are the undecidable properties in all axiomatic for formalizing the arithmetic.

KEYWORDS: Turing, intuitive sense, arithmetic, incompleteness, artificial intelligence.

RESUME: Le degré de liberté de la machine est organisé et spécifié par l'homme. Quand bien même la machine dépasse l'homme en rapidité et en fiabilité, celui-ci reste le maître de l'algorithme et marque dans la conception de l'algorithme sa supériorité vis-à-vis de la machine qui ne doit qu'exécuter l'algorithme qui a été prévu. Les fonctions calculables à la TURING sont toutes calculables au sens intuitif. Ainsi, il y a des propriétés indécidables dans toute axiomatique pour formaliser l'arithmétique.

MOTS-CLEFS: Turing, sens intuitif, arithmétique, incomplétude, intelligence artificielle.

1 INTRODUCTION

En 1930 et 1931, GÖDEL a atteint des résultats fondamentaux concernant la problématique de la décidabilité en logique et en mathématiques. On dit en logique qu'une théorie présente une incomplétude si elle comporte une formule telle qu'elle ne peut être ni démontrable ni réfutable, donc indécidable.

En effet, le théorème de GÖDEL affirme, d'une part, l'existence de formules vraies dans la structure (N, O, S, +,.) et prouvables à partir du système de PEANO de premier ordre PA_1, et, d'autre part, l'impossibilité pour une théorie consistante incluant PA_1, de prouver sa propre consistance par des procédés formalisables, c'est-à-dire la cohérence de l'arithmétique de PEANO. Grâce à une extension de la théorie des fonctions récursives, ce théorème peut être généralisé de diverses façons. L'étude des fonctions récursives constitue une voie d'approche permettant de mieux cerner le contenu de la notion de constructivité.

Pour GÖDEL donc, aucun système de preuve ne peut capturer pleinement le raisonnement mathématique. En clair, GÖDEL affirme que toute théorie suffisante, aux fins de capturer les raisonnements arithmétiques, est nécessairement incomplète. Il convient de souligner qu'il y a des énoncés non prouvables et dont la négation n'est pas non plus prouvable. Donc la cohérence d'une théorie mathématique s'exprime par un énoncé, qui ne peut être prouvé ou réfuté. Au fond, les arguments gödeliens sont fondés sur une notion informelle de déduction algorithmique.

CHURCH, pour sa part, a résolu de façon négative, l'Entscheidungs Problem. De son coté, TURING a atteint un résultat de portée fondamentale, connu sous le nom de machine de TURING.

2 SYSTEMES FORMELS ET MACHINE DE TURING

HILBERT explique la nature discrète des formalismes mathématiques qui sont considérés comme des suites des signes matériels simples ou élémentaires et alphabétiques. Donc, une fois qu'on voit les lettres et les mots gödelisés, la voie est ouverte à la machine numérique de TURING. Pour Giuseppe LONGO, du moins au regard des structures conceptuelles des mathématiques, des systèmes formels alphanumériques sont à nouveau supposés complets. Cela veut dire donc que des systèmes formels auraient dû nous dire tout de ces structures conceptuelles. Et, au vu des certains, la machine de TURING aurait dû un jour modéliser complètement le fonctionnement du cerveau. A chaque fois qu'il se montre audacieux, le programme de connaissance paraît se dégrader toujours plus.

La machine de TURING ou machine logique s'inscrit dans le cadre des systèmes formels aux dynamiques continues. Cette démarche est en effet « Laplacienne ». Cela signifie donc que le caractère imprévisible ne peut être que pratique et, en principe, n'existe pas comme dans la physique des systèmes formels[1].

Profitant des résultats obtenus par GÖDEL, CHURCH a donc utilisé sa propre formalisation liée à la notion intuitive d'algorithme. Finalement, il a abouti à un résultat fondamental suivant : « λ −calcul ».

	S_1	S_6	S_3	S_2	S_{10}	S_{22}	
			$\uparrow$				
			q_7				

Notons en effet que la machine de TURING est un dispositif abstrait qui se compose d'une bande infinie divisée en cases, et d'une cellule mobile observant à chaque moment une de ces cases. La cellule contient le programme de machine de TURING, c'est-à-dire l'ensemble des instructions.

Il est à noter que chaque instruction prend la forme d'une suite ordonnée de 5 symboles. Il s'agit de S_i, q_1, S_j, q_m, X. Cela signifie que si la machine de TURING est à l'état q_1 , et si le signe S_i est dans la case visée, alors elle doit effacer la cellule d'une case vers la gauche ou vers la droite ou encore la laisser à sa place, selon la valeur de $X(X = G: gauche; X = D: droite: X = N: Nul)$ et, finalement, passer à l'état q_m. L'ensemble fini de signes $\{S_1, S_2 \dots S_k\}$ est l'alphabet extérieur de la machine de TURING et, sert, dans le cadre du programme, à diriger le travail de la machine. TURING tire deux propriétés de base de son modèle : l'universalité et les problèmes indécidables[2].

En effet, la puissance du modèle de TURING s'exprime dans l'universalité. Au fond, elle est la propriété la plus prodigieuse. L'universalité s'énonce intuitivement comme suit : il existe une machine de TURING U qui, à partir d'une description d'une autre machine de TURING M et d'une donnée de M, va calculer la même chose que M. En clair, U *simule* M. TURING a montré comment il était possible de construire une « Machine de TURING universelle » qui puisse être en mesure de stimuler l'action de n'importe quelle machine de TURING équipée de n'importe quel programme ; on l'appelle U. Elle a un programme fixe. Les données initiales de la machine U sont une suite de 0 et de 1 ayant la forme code (M).d pour stimuler une machine disposant du programme Met traitant des données d. En effet, le programme de la machine U procède à l'analyse du code du programme M, et il applique ce programme aux données d [1].

Pour TURING, le $\lambda -$ calcul typé obtenu par la suite était expressif. Cela signifie donc qu'on a au plus les polynômes. Ce qui nous écarte à cette époque de la fonction d'ACKERMANN.

---

[1]*La machine de TURING est un objet théorique consistant en trois éléments : un ruban, une tête de lecture/écriture dite simplement tête de la machine et une liste d'instructions.*
TURING, en 1936[-], considérait en effet l'homme calculant, le computeur, et se demandait comment décrire systématiquement et exhaustivement son activité mentale idéalisée. L'auteur, en 1950[1], présentait autrement sa préoccupation. C'est toute l'étendue des activités ou manifestations de l'intelligence qu'il prétend ici subsumer et cela sous un schéma explicatif général.
[2] *M. MARGENSTERN, "Ce que Alan Turing nous a laissé" dans SMF – Garette 135, janvier 2013, pp.17-31.*

La formalisation de sa machine permet à TURING de donner une réponse à un des problèmes majeurs posés par HILBERT. Il y répond donc par la négative. D'un point de vue théorique, on peut penser que tous les calculs sont réalisés sur une machine de TURING et de définir un algorithme qui peut être un programme de TURING-POST [1].

En clair, TURING prouve qu'il n'existe pas d'algorithme qui permette de résoudre le problème de la décision.

Autrement dit, il n'y a pas d'algorithme permettant de savoir si un énoncé clos du calcul des prédicats du premier ordre est démontrable ou pas. A ce stade, il évoque le problème de l'arrêt des machines de TURING. Sur ce, on dit que le problème de l'arrêt est *indécidable* puisqu'il n'admet pas l'algorithme de décision. Cela amène à un théorème prouvé par RICE mais que TURING aurait pu démontrer à son niveau. Il s'agit d'un nombre réel récursif en une suite de 0 et de 1 définie par une machine de TURING. Ici, on donne n par exemple à la machine ; celle–ci retourne les n premiers termes de la suite. Ainsi, le réel représenté est $\sum_{k=0}^{\infty} a_k 2-k$ où a_k est la suite fournie par la machine. La propriété est donc la suivante : *il n'existe pas d'algorithme permettant de dire si un nombre réel récursif est nul on non*.

Par la suite, TURING a abordé le $\lambda-$calcul de CHURCH. Devant la difficulté créée par la non–totalité d'une fonction universelle, CHURCH et TURING ont eu à transporter au $\lambda-$calcul la notion russellienne de type.

En 1950, on reconnaît les efforts entrepris par GÖDEL avec son système T[3] qui, tout en restant dans un ensemble de fonctions totales, admet un schéma de récurrence fonctionnelle permettant de passer au – delà des fonctions primitives.

Le problème de l'arrêt étant un problème indécidable, il a été ressenti comme une barrière difficile à franchir. Ce problème consiste à déterminer quels jeux de données vont déterminer ou non l'arrêt. Ce serait intéressant que le problème de l'arrêt puisse se résoudre par un algorithme, c'est-à-dire par la recherche d'un programme de TURING-POST. On doit à TURING une extension de la notion de machine de TURING appelé *machine à oracle*. En effet, on dit que la machine à oracle est une machine de TURING ordinaire qui dispose d'un ruban supplémentaire ayant une tête de lecture seulement avec ce ruban. Il résulte de l'indécidabilité du problème de l'arrêt l'existence de limitations internes qui détermineraient l'applicabilité des algorithmes. En fait, la preuve constitue une application de l'argument diagonal de CANTOR.

Evidemment, elle calcule comme une machine de TURING ordinaire. Elle peut parfois consulter le ruban supplémentaire appelé oracle. Dans une telle démarche, un pas de calcul va compter.

La nouvelle puissance des machines à oracle dépend effectivement de l'oracle en ce sens que si on fixe un oracle, on a immédiatement un nombre infini de machines. En clair, toutes les machines de TURING ordinaires pourraient être adaptées aux fins de fonctionner avec un oracle.

3 ESPRIT HUMAIN ET MACHINE

Le lien qu'établit TURING entre systèmes formels et machines permet l'application du résultat de GÖDEL à l'intelligence artificielle. En 1936, TURING invente la notion de machine de TURING, procédé formel qui permet de définir ce qu'est un énoncé décidable ou une fonction calculable. Avant même la naissance officielle de l'intelligence artificielle, sa possibilité théorique est donc mise en cause par le théorème de GÖDEL illustrant sans contexte les limitations de l'approche mécaniste.

En montrant que les machines ne peuvent pas tout calculer, les travaux de GÖDEL et de TURING[4] paraissent s'opposer au projet de l'intelligence artificielle. Or, pour TURING, il n'y a pas de limites entre « mécanique » et « intelligent » et le cerveau est semblable aux ordinateurs en train de naître. Il estime que les machines vont devenir « intelligentes ». Ainsi, en 1950, il propose un test qui permettrait de déterminer quand cela sera atteint[5]. Cette vision de la pensée humaine est perçue comme

[3] *Le système T de GÖDEL est le système construit lorsqu'on ajoute au $\lambda-$ calcul simplement typé les types produits, les types sommes, le type unité, le type vide et le type des entiers naturels. En effet, ce système où la réduction est confluente et satisfait la propriété de CHURCH – ROSSER – vérifie les propriétés de déclaration des variables libres, d'affaiblissement, de substitutivité de préservation du type par réduction et même d'unicité du type (sauf dans la présentation à la CURRY).*
Evidemment, l'inférence et la vérification dans le système de T de GÖDEL sont donc décidables. Finalement, on voit que le système T vérifie la propriété de normalisation forte entrainant que tous les programmes bien typés dans ce système formel terminent.

[4] *TURING, « Computable Numbers, with an Application to the Entscheidung Problem » TURING montre l'indécidabilité du problème d'arrêt d'une machine de TURING en utilisant, comme GÖDEL, la méthode de la diagonale de CANTOR.*

[5] *A. TURING, « Computing Machinery and Intelligence »*

un mécanisme mathématique. Dans le test de TURING, l'auteur critique toutes les objections faites à la possibilité d'une machine pensante (théorème de GÖDEL, la conscience, l'objection d'ADA LOVELACE, etc.) et les réfute.

La question du rapport entre le théorème de GÖDEL et l'intelligence artificielle nous invite à préciser quelques nuances. [2]

Nous estimons qu'il est possible de mécaniser l'activité de certaines zones du cerveau. Cela veut dire qu'on peut automatiser la reconnaissance liée à l'espace et fabriquer ainsi des robots qui peuvent retomber sur leurs pattes. De même, ils peuvent corriger par exemple les déviations de la trajectoire d'un véhicule au conducteur qui somnole. S'agit-il ici d'intelligence ou d'instinct qui guiderait le robot ? Quant à nous, il convient de parler d' « instinct artificiel ».

En 1950, TURING s'efforçait de montrer qu'il était concevable que l'on puisse dire un jour de certaines machines qu'elles sont capables de penser ou encore de faire preuve d'intelligence. Dans son schéma, la stratégie de TURING comportait deux volets [3] :

1° le volet conceptuel et négatif qui estime faux d'attribuer de l'intelligence à une machine quelle qu'elle soit.

2° le volet empirique et positif qui fait valoir la plausibilité de l'hypothèse que les ordinateurs soient un jour capables d'un comportement intelligent.

On lit bien les tensions chez TURING dans les deux moments de sa recherche. De nature mathématique, l'article de 1936 a entre autres indiqué les limitations internes du formalisme à la HILBERT. De portée philosophique, celui de 1950 a circonscrit aussi, contrairement aux anciens préjugés, que le domaine de la machine s'étendait potentiellement bien au–delà de l'exécution des tâches « mécaniques » - au sens de « stupides » qui n'impliquent ni pensée ni volonté – et cela, de façon concomitante faisant voir que le formalisme était plus la pure pensée mathématique. C'était en effet déjà là l'annonce du programme relatif à « l'intelligence artificielle ».

Les machines réalisent leurs opérations souvent plus rapidement et de manière plus fiable que l'être humain par l'exploitation efficace de plus grandes informations, et cela, quand bien même elles s'évertuent à pouvoir réfléchir sur ce qu'elles sont en train d'effectuer et à pouvoir faire progresser leur comportement.

En tout état de cause, quelles que soient les performances, on ne peut pas se permettre d'affirmer qu'elles sauront résumer les textes et être en mesure d'appréhender des émotions, des intentions. Elles ne pourront pas non plus évaluer des faits liés à l'esthétique. En bref, tous ces faits relèvent de la sensibilité propre à l'être humain.

Evidemment, il y a lieu de souligner que la machine ne fait que résoudre des problèmes qui sont souvent difficiles, et que l'être humain lui pose. Au fond, ils sont en effet exprimables dans un cadre formel éventuellement très général. Donc, on ne pourrait pas affirmer que la machine pense à la manière de l'être humain, car la pensée ne se conçoit pas sans la conscience d'elle–même.

La machine sert donc de support dans les activités de l'homme. Il est donc difficile de discerner les limites intrinsèques de l'interaction ou association homme–machine. Toutefois, on retiendra que la machine reste, du point de vue mécanique, un outil surpuissant au service de l'homme.

Finalement les machines sont perçues comme de complexité limitée. Elles sont donc incapables de sortir des comportements spécifiés par leurs concepteurs. Ainsi la réalité des machines est telle que les ordinateurs modernes conduisent forcément à reconsidérer cette conception.

Il sied de souligner que l'intelligence ou la créativité relève d'un domaine plus complexe. Donc, on peut dire que c'est voué à l'échec. En clair, le théorème de GÖDEL s'y oppose radicalement. Il réfute la considération d'une science mécanisable et finale. Il vaudra mieux utiliser le bon sens. Par ailleurs, l'intelligence comprise au sens créatif se situe en une position d'inconfort où la possibilité d'erreurs. Donc, ici *le droit à l'erreur* se présente comme la condition *sine qua non*.

Nous devons en effet éviter le simplisme et de faire des exagérations. L'intelligence humaine atteint le pouvoir suprême et peut faire « bénéficier le monde des errements de la pensée, ce qu'un robot ne saurait faire. Nous précisons alors que « par rapport au fantasme d'un robot-Einstein, cela peut sembler limité, mais la véritable ambition se pare des atouts les plus modestes » [3].

En vertu des théorèmes d'incomplétude de GÖDEL, une machine qui peut appliquer des règles d'inférence sur un nombre fini d'axiomes ne saurait se prononcer sur la vérité de certains énoncés : les phrases de GÖDEL. Pourtant, on sait que l'être humain peut se rendre compte que ces énoncés sont vrais.

On reconnaîtra que les théorèmes d'incomplétude ont la forme conditionnelle : si la théorie et son implémentation dans la machine sont consistantes, alors l'être humain est capable d'affirmer la vérité de G. Pourtant, l'esprit humain éprouve d'extrêmes difficultés à déterminer si une théorie est consistante, et donc la méthode pour y arriver n'existant pas.

Cela suppose que l'être humain peut déterminer la consistance de toute théorie, ce qui nous laisse plonger dans la pure spéculation.

Il convient de souligner que l'affirmation selon laquelle l'arithmétique est indécidable, signifie donc qu'il n'existe pas de canaux automatiques de savoir, si une formule donnée est vraie ou pas pour les nombres naturels.

Au-delà d'un problème mathématique, les théorèmes d'incomplétude posent un problème scientifique beaucoup plus général. Ils montrent en effet qu'une théorie scientifique peut expliquer l'ensemble des phénomènes observés et que deux théories radicalement opposées, peuvent coexister et s'appliquer aux mêmes phénomènes naturels. A ce sujet, la coexistence de la physique relativiste et de la mécanique quantique en illustre l'exemple le plus frappant.

4 CONCLUSION

D'après TURING, la classe des fonctions à la CHURCH était équivalente à la classe des fonctions programmables sur les machines imaginaires ou machines de TURING. Il existe une procédure de calcul assimilant les fonctions calculables aux fonctions exécutables sur une machine de TURING. En réalité, les fonctions calculables à la TURING sont donc toutes calculables au sens intuitif.

Toutefois, il existe de propriétés indécidables dans tous les systèmes formels ou axiomatiques pour formaliser l'arithmétique. De façon correcte et évidente, il y a lieu de dire qu'on ne peut pas concevoir un algorithme qui, étant donné un programme quelconque, avertit avec certitude et au bout d'un temps fini, si ce programme doit éventuellement « tourner en rond », et s'exécuter de manière indéfinie, sans jamais trouver la sortie. Ainsi, il est établi qu'à toute propriété mathématique, on peut toujours associer une fonction numérique et réciproquement, à toute fonction numérique, il est possible d'associer une propriété mathématique. Donc, il faut remarquer qu'une propriété mathématique P(a) est une fonction vers un ensemble ayant deux valeurs, {vrai, faux} ou, équivalent, {0,1}. D'où, le problème de la décision est équivalent au problème de calcul.

REFERENCES

[1] Ian, STEWAERT : *Les mathématiques*, traduit de l'anglais par François GALLET, (Collection Sciences d'Avenir), Paris : la science, 1989.
[2] J-Y. GIRARD : *le point aveugle*, tome I, vers la perfection, Herman.
[3] F. NEF et D. VERNANT (dir) : *Les années 1930 : réaffirmation du formalisme*, Vrin, pp.1-41, 1998.