

## **Joli nouveau monde** **Stratégies dans la société numérisée**

*Sous ce titre eurent lieu du 18 au 19 octobre 2019, au Forum 3 de Stuttgart, un atelier public et un colloque de recherche. Ils furent organisés par l'Institut pour les questions sociales du présent e.V. en association avec l'Initiative Réseau Dreigliederung. Les articles qui suivent de Han –Florian Hoyer et Stefan Padberg s'édifient sur leur contributions respectives à ce forum ; Han-Florian Hoyer dirigea la première partie de la discussion le samedi, Stefan Padberg la conférence de la veille au soir. L'article de Kai Ehlers, sur le défi du « transhumanisme », se rattache par son contenu à un sujet qui va être traité en propre une fois encore en mai prochain lors d'un colloque (voir les annonces dans ce numéro en dernière page).*

### **Crépuscule de l'être humain dans la machine, crépuscule de la machine chez l'être humain** **Hans-Florian Hoyer**

La différence la plus nette entre le modèle de Descartes<sup>1</sup> et la présentation de Fritz Kahn<sup>2</sup> consiste dans le fait que Kahn attire les choses du monde créés par l'être humain et les transpose de manière imagée dans son corps. Un cinéma dans la tête et un orgue dans le larynx. Dans ce sens on peut parler d'un « crépuscule des machines dans (l'image de) l'être humain ». Il éclaire le prodige du corps avec des choses connues. À cette occasion le contenu de réalité de l'analogie reste quelque fois sur le carreau.

Au moment où Kahn fut interpellé par un ami suisse au sujet de quelques fautes dans son ouvrage « *L'atome* », il a dû répondre seulement de manière lapidaire : « Eh bien ! en effet, il est déjà faux, mais compréhensible ! »

On procède exactement à l'inverse, pour expliquer la machine aux êtres humains qui se fait toujours plus remarquer dans son quotidien : l'ordinateur. Ici des analogies sont tirées du monde corporel, pour expliquer une technique avec laquelle rien ne s'extériorise ni ne se meut, ou selon le cas, rien des processus de calcul n'est à voir.

Cela ne veut pas dire : « l'ordinateur, la machine que tu n'as pas encore comprise », mais plutôt « l'ordinateur, la chose inconnue ». Qu'est-ce qui a poussé quelqu'un à réaliser cette mise en page ? [Une image d'un ordinateur sortant d'un œuf à la coquille brisée à l'instar un petit poussin, *ndt*] Quelqu'un qui connaît vraisemblablement foncièrement la technologie du *hardware* et *software*. Mais qu'est-ce qui l'a poussé à le faire sortir d'un œuf de poule à l'instar d'un poussin ?

Parfois il est « cœur » : « le processeur — le cœur de l'ordinateur », parfois il est « cerveau » : « le microprocesseur est pour ainsi dire le cerveau de ton ordinateur étant donné qu'il mène et gouverne toutes les opérations de l'ordinateur. »

La mémoire est celle de ses informations et la communication avec cette chose s'effectue par un « langage ». « Nerfs » et électricité élargissent les possibilités de tirer d'autres faux parallèles de ce genre.

#### **Le penser s'obscurcit lui-même**

Avec une programmation correspondant et des données remplies de sens, les machines désignées comme « ordinateurs », peuvent fournir foncièrement des résultats sensés. Pour quelqu'un qui embrasse du regard et comprend leur processus de construction, la programmation, ainsi que les données, les ordinateurs ne sont plus guères étranges. Chez quelqu'un pour qui ce sont des « boîtes noires » dont les contenus lui sont expliqués par des analogies, l'impression peut alors surgir que certes, de l'intelligence est ici en jeu, puisque le résultat en éveille un semblant. L'erreur repose dans le fait que l'intelligence réellement en jeu ici, n'est pas visible, c'est celle qui a été dispensée par les constructeurs, ceux qui ont projeté le système, les inventeurs des langages de programmation et finalement aussi celle des programmeurs. Toutes ses intelligences se présentent sous la forme d'une essence et elles n'avaient besoin que d'être « appelées » — quand on le veut et aussi souvent qu'on le veut. On est donc enfin parvenu à formaliser une partie de la

<sup>1</sup> [Il s'agit de deux transpositions de « l'homme-machine », *ndt*] [https://archive.org/details/bub\\_gb\\_Tiv\\_Wt6uac0C/page/n111](https://archive.org/details/bub_gb_Tiv_Wt6uac0C/page/n111)

<sup>2</sup> <https://www.speigel.de/geschichte/koerper-erklaerer-fritz-kahn-a-946748.html>

pensée humaine et à construire une machine qui peut en exécuter ses formes, comme le peut aussi tout être humain vivant, quand bien même encore plus lentement et avec moins de sûreté.

Cette auto-réflexivité du penser, qui peut mettre en place la règle de son propre usage (!), commença avec Aristote, se contraignit, avec la scolastique, en chaussant les « bottes espagnoles » du *Collegium Logicum* et se déploya en passant par Raimundus Lullus, Leibnitz, Frege, Boole, pour former la base d'une externalisation d'un domaine de la pensée humaine. Cela a commencé tout petitement à l'instar de rechercher un valet pour effectuer des calculs : comme des exécutions réitérées des mêmes opérations de calcul avec d'autres valeurs des chiffres.

Wilhelm Schickard eut pitié de Johannes Kepler, Blaise Pascal de son père, le percepteur supérieur des impôts de Normandie, et ils tentèrent tous deux s'externaliser le penser pour la nécessité du calcul. Chez Konrad Zuse, ce fut ensuite l'intelligence appariée à la paresse (« *J'étais tout simplement trop fainéant pour compter.* », expliqua-t-il) qui l'amènèrent à développer le Z 3 pour résoudre des tâches relevant de la statique des bâtiments.

Leibnitz s'essaya aussi à la réalisation d'une machine à calculer, mais il alla bien au-delà du calcul avec sa *lingua characteristic* et le *calculus ratiocinator*. Si tous les principes valables des faits du monde étaient formulés sur lesquels tous les philosophes d'accord sur la *lingua*/langue, on pourrait abandonner des valets de calcul et au moyen du calcul en retirer d'autres conclusions valables (*calcuemus*). Toute querelle serait superflue, on pourrait compter pour savoir si quelque chose est juste, c'est-à-dire si cela convient aux principes reconnus.

Dans le *calculus* on opère avec des symboles — tout indépendamment de la connaissance de leur signification. Ces opérations ne nécessitent chez l'opérateur, ni une compréhension du contenu pas plus qu'elles ne font naître celle-ci par l'opération. Pour quelqu'un qui en connaît la signification elles font naître l'apparence que celui qui opère a aussi une compréhension, en effet, il doit l'avoir, sinon comment pourrait-il opérer d'une manière aussi intelligente ? John Searle a décrit pour cela une expérimentation.<sup>3</sup>

#### **Intelligence naturelle : vivre, apprendre et découvrir**

Une intelligence se révèle chez des êtres vivants au comportement et dans la communication. En dehors de l'être humain, on peut l'observer chez les animaux et les végétaux. Pour pouvoir conférer de « l'intelligence » aux comportements des animaux et des végétaux, l'observateur humain doit pouvoir coordonner un sens à ce comportement.

Des plantes qui « avertissent » leurs voisins d'une attaque de nuisibles, puis réagissent pour leur part avec la sécrétion de substances de défense, se situent donc dans un contexte de comportement que l'on peut suivre par la pensée et estimer comme rempli de sens. L'attribution d'une intelligence conditionne additionnellement des connaissances sur la manière dont un tel comportement a pris naissance. S'il s'agit d'une sorte « d'automatisme encastré dans la structure édifiée », on devrait plutôt parler alors d'une construction de la nature pleine de « sagesse ».

Pour la vie du porteur et de son comportement rempli de sens, se rajoute encore la nouvelle manière de se comporter découverte spontanément par lui et d'anciennes qui peuvent être modifiées sur la base d'une expérience réalisée.

#### **Intelligence artificielle : pas de vie et la simple apparence d'apprendre, de découvrir, etc.**

L'expressions « d'intelligence artificielle » est une manière abrégée de parler pour : « un semblant d'intelligence qui a été appelée à naître de manière artificielle, sans qu'il y ait à sa base un être naturel conscient. »

Un observateur qui ne connaît pas les possibilités des automates et leur programmation, peut y gagner l'impression de se trouver en face d'un instrument « doté » de l'intelligence. Au début de développement de cette technique d'intelligence, ce fut la vitesse d'action de la machine, qui fut exploitée de manière décisive pour la mise au jour de l'apparence, du semblant, entre temps des mémoires presque illimitées sont venues

<sup>3</sup> [https://de.wikipedia.org/wiki/Chinesisches\\_Zimmer](https://de.wikipedia.org/wiki/Chinesisches_Zimmer)

s'y adjoindre en plus. Et aussi la communication qui s'échappant de la contrainte du clavier et du texte sur l'écran, y contribua en s'émancipant par surcroît, en langage et image.

À une construction matérielle qui ne vit pas, on ne peut nonobstant pas attribuer d'intelligence propre, parce que le comportement des constructeurs (intelligents !, eux) est fixé, déterminé, par le *hardware* et le *software*. En outre, la signification des modèles d'octet dans l'unité centrale et de la mémoire est fixée par des conventions. Un extrait quelconque n'en est pas lisible pour des êtres humains ne disposant pas d'informations additionnelles, quant à savoir s'il s'agit de texte, image, son, commandes, nombres ou autres. Le silicium n'a que les états binaires et ne connaît pas leur sens.

**L'utilisation du concept d'intelligence dans les deux cas est une malpropreté du penser qui se venge, celle-ci, lorsqu'on l'oublie.**

### **Architecture du calculateur**

Le calculateur consiste en transistors reliés les uns aux autres, autrefois des relais/diodes, qui peuvent avoir deux états binaires discrets. Par les conducteurs électriques du « bus », comme on l'appelle, des impulsions cadencées sont envoyées qui doivent toujours être de nouvelles réitérées. Dans, l'unité arithmétique-logique (*ALU : Arithmetical Logic Unit*), les instructions qu'elle peut exécuter, sont mises en œuvre ensemble avec les données qui proviennent d'une mémoire [morte/vive, *ndt*] [ou « magasin d'informations stockées », d'où le mot allemande de *Speicher* =, grenier magasin, entrepôt]. La « lecture » d'une telle « mémoire » consiste dans l'instauration d'une copie des modèles d'octet qui y existent. La mémoire pour les instructions n'est physiquement pas séparée de celle des données (architecture de von Neumann). Aussi bien la prochaine instruction comme la prochaine donnée doivent arriver à l'*ALU*. Le résultat doit y être déposé dans la mémoire de travail.

Von Neumann s'est résolu à ne pas séparer physiquement données et programmes dans deux mémoires distinctes. Ainsi la possibilité fut conservée d'exécuter des instructions qui de leur côté étaient stockées dans la mémoire et pouvaient aller y être recherchées plus tard afin de les représenter. Et donc des programmes qui furent rédigés pour écrire des programmes. Ainsi le semblant d'un apprentissage a-t-il été engendré.

### **Langages de programmation**

Enfin, les instructions doivent chargées en mémoire sous forme binaire, l'adresse de début dans le processeur et un bouton de démarrage doit être pressé. Le processeur « va chercher » l'instruction, l'exécute et relève l'adresse pour la prochaine de « 1 », s'il ne devait pas exécuter un saut de commande : l'adresse est alors déterminée autrement pour la prochaine instruction. Normalement les instructions sont chargées de manière séquentielle.

Au lieu de modèles binaires, les programmes sont écrits en assembleur, le « langage » de programmation le plus proche de la machinerie. À partir d'un programme, de l'assembleur, les lettres « NOT », « AND », « XOR », ... sont transposées dans le modèle binaire correspondant. Une programme de chargement les copie ensuite aux endroits corrects et transmet le contrôle à l'unité centrale du processeur (*CPU : Central Processing Unit*). Les premiers « langages » orientés sur des problèmes étaient le FORTRAN et COBOL. Dont les phrases d'instruction étaient transposées dans l'assembleur par un compilateur et ensuite de nouveau comme ci-dessus. Aujourd'hui nous connaissons le C++, Java, Python. Ce dernier charge chaque ordre mais toujours dans une unité de modèle binaire ou plusieurs.

Dans ce contexte parler de « langage [on devrait dire « encodage » en français, *ndt*] » c'est un des péchés originels du « crépuscule de l'être humain dans la machine ». Ce serait plus adapté de dire « liste d'appel d'une cour de caserne ».

### **Apprendre**

Un calculateur, dans lequel le programme est chargé, dont le résultat est la communication d'une imposition fiscale d'un mandant, n'a jamais « appris » celle-ci, pour passer sous silence ce qu'il « sait » de ce qui se passe lorsqu'il « déroule » le processus. Qu'un résultat, interprétable de manière sensée pour un être humain, soit engendré, cela n'a à faire qu'avec l'intelligence des êtres humains qui ont projeté le *hardware*, ainsi que celle des programmeurs qui ont « saucissonner » le processus en étapes, le cheminement de la résolution du problème à formaliser en le rendant compatible avec les instructions qui sont formulables et qui appartiennent à l'unité centrale du processeur.

Or l'apprentissage, ce n'est pas un simple emmagasinage de contenus. Si la « mémoire » des calculateurs fonctionnaient comme la nôtre, nous la considérerions comme « déficiente » [*kaputt* = foutue en « allemand » *ndt*]. Une expérience n'est pas l'histoire des processus de mises en mémoire informatique. C'est une tâche passablement intéressante d'écrire un programme qui évalue les séquences des états d'un jeu sur un échiquier et ses évaluations (perte/gain) et les transpose en étapes de programme qui à la suite de très nombreuses étapes d'analyses, éveillent l'apparence que le calculateur eût « lui-même » appris le programme pour l'exécuter parce qu'en définitive, on n'en a pas programmé les règles. C'est comme si Mouret, Allgeier et Schlumberger avaient formé les Turcs jouant aux échec, engagé de jeunes joueurs en herbe et dans le même instant oublié qu'ils ont fait cela.

### **Un robot se reconnaît dans un miroir**

« Oh ! Cela, je suis en effet. Beau », dit le robot *Qbo* (à prononcer « cubio » !) après qu'on lui « eût expliqué » que « c'était « lui-même » qu'il voyait dans le miroir ».

Ce texte est solidement encodé d'avance dans le programme. *Qbo* « synthétisera » à l'avenir à chaque fois ce même texte, de sorte qu'il en naîtra l'apparence qu'il parle. La condition c'est qu'un modèle qui est solidement chargé dans sa mémoire est comparé à un modèle dans un miroir et qu'il en surgisse une conformité exacte. En outre le code de programmation doit être présent qui puisse classer un recoupement partiel d'avec l'image de son entourage comme miroir. Une tâche de programmeur intéressante.

### **Conduite autonome**

Une automobile se meut de manière autonome, lorsqu'on n'en voit pas le conducteur [C'est la raison pour laquelle il est interdit en France de teinter les vitres des portières avant et du pare-brise, car on ne peut plus identifier le conducteur. *Ndt*] On ne peut parler de conduite autonome que lorsqu'on fait disparaître l'électronique de la conduite.

Parler « d'apprentissage autonome », cela signifie qu'on oublie qu'un programmeur a mis sa représentation de cet apprentissage en une succession d'instructions qui peuvent se dérouler comme autant de modèles binaires sans qu'un « soi » ait la conscience de ce qui se passe. En cas de réussite pour le programmeur cela peut engendrer un semblant d'intelligence pour un observateur qui n'en connaît pas les processus, qui n'a aucun aperçu dans la boîte noire qui est au cœur du système [Jusqu'au jour n'ayant pas correctement programmé la forme d'un cycliste, la voiture a percuté une authentique dame qui faisait authentiquement du vélo et qui a été authentiquement tuée. *Ndt*] L'**ia** (intelligence artificielle) est toujours un semblant d'intelligence pour un observateur qui n'a pas cet aperçu dans la boîte noire.

### **Test de Turing**

Alan Turing évita le problème de ce qu'on veut dire ici par le terme « d'intelligence ». Il préféra interpréter ce mot en ayant recours à un subterfuge expérimental. Il plaça un être humain devant deux écrans avec un clavier. Un écran est relié à un ordinateur, l'autre à un autre écran et clavier utilisés par une autre personne, dans un local caché. Le premier est censé découvrir au moyen de questions ciblées lequel des deux postes d'écran est géré par l'ordinateur et lequel l'est par la personne humaine cachée. Turing affirme que l'on parle d'intelligence artificielle lorsque l'être humain qui réalise ce test ne peut plus déterminer lequel des deux écrans est relié à l'ordinateur.

Quelles seraient les questions qui révéleraient cela ?

- **Quelle est la racine septième de  $\pi$  ?**

L'écran sur lequel apparaît par retour du message le nombre « 1,1776640300 » est relié à l'ordinateur. Pour éveiller ici le semblant d'intelligence il faut programmer d'avance artificiellement un temps de réflexion conséquent. Mais pas par exemple pour la racine cubique de 8, car ça c'est trop facile !

- **Poser des tâches de calculs une heure durant.**

Celui qui fait des erreurs, c'est l'être humain. Pour que cette possibilité de reconnaissance soit exclue, il faut programmer artificiellement des erreurs.

- **Taper un histoire**

*Un Texan se pointe dans une taverne de la Frise orientale et dit devant deux Frisons qui sont en train de boire une bière : « J'ai une ferme gigantesque au Texas ». Les deux Frisons se regardent et l'un dit : « Ici aussi, nous avons aussi ». Le Texan dit : « Il me faut une journée pour faire le tour de ma ferme en auto. » Les deux Frisons se regardent et l'un dit : « Une telle auto, nous avons aussi ici. » Si sur l'écran n'apparaît pas passablement vite « Bonne blague » ou « bien « je la connaissais déjà », alors ce n'est pas l'être humain. L'humour est passablement difficile à programmer.*

### **Résumé**

Comment l'être humain peut-il conserver sa souveraineté vis-à-vis des machines et les résultats que produit les programmes qui ont été démarrés en elles ?

Une supposition inconditionnelle, c'est la vérification des résultats quant à leur plausibilité. Cela vaut pour un programme d'impôts sur les salaires, comme pareillement pour un programme de calcul statistique des ponts et chaussée. Autrefois on les faisait calculer par deux architectes ou respectivement deux programmes différents pour la construction d'un pont.

Le scandale au sujet d'un *software* dont on a négligé la vérification pour le *boeing 737 Max* rend évidente l'importance de la certification des programmes dont des vies humaines dépendent de leur fonctionnement correct.

Des programmes sont de plus en plus mis en œuvre pour instaurer des pronostics, à partir desquels ensuite des actions sont menées. Quelqu'un qui vit dans la même rue d'un terroriste recherché est un complice éventuel avec une certaine vraisemblance. Lorsqu'une évaluation d'intelligence technique est prise de manière trop absolue pour des êtres humains qui ne voient que l'hameçon « **de soupçon de terrorisme** », sans savoir pourquoi un tel programme a été mis en œuvre, il peut en résulter des désagréments pour la personne en question.

Entre temps il existe des programmes qui repèrent les traces laissées inéluctablement derrière elles par des personnes dans ce qu'on appelle les réseaux sociaux ou sur *Internet*. Ceci décrit la douzaine d'êtres humains qui se dissimulent parmi nous, plus précisément qu'il nous est éventuellement conscient de le faire.<sup>4</sup> N'y a-t-il pas là-dedans le défi d'édifier un individualisme authentique ?

**Sozialimpulse** 4/2019.

(Traduction Daniel Kmiecik)

**Hans-Florian Hoyer** : a suivi de près l'entrée dans la technologie informatique au quotidien à la fin des années 1970. Il a décrit en profession libérale des programmes *softwares* pour calculateurs de poches programmables, l'*Apple II*, et les « ordinateurs personnels » qui ont suivi, ainsi que pour de plus grands calculateurs pour des bureaux d'ingénieurs. Depuis 1986 il est régulièrement sur *Internet*.

---

<sup>4</sup> <https://t3n.de/news/besteller-autor-meint-algorithmen-werden-uns-besser-kennen-als-wir-selbst-1161597/>