Alan Turing et l'intelligence artificielle : le " jeu de l'imitation " et " l'IA forte "

Patrick Goutefangea

Alan Turing et l'intelligence artificielle : le « jeu de l'imitation » et « l'IA forte »

Patrick Goutefangea

I

Dans *Computing Machinery and Intelligence*[1], publié en 1950 dans *Mind*, Turing examine la question « Les machines peuvent-elles penser ? » (*« Can machines think ? »*), et propose de la remplacer par un test, le fameux « jeu de l'imitation ». Celui-ci oppose un joueur C – un homme ou une femme - à deux partenaires, A – un homme - et B – une femme. C est isolé des deux autres. Il doit, en posant des questions à A et B, déterminer qui est l'homme, qui est la femme. L'homme A doit s'efforcer de le tromper en se faisant prendre pour la femme B, laquelle doit, au contraire, l'aider. Les trois protagonistes ne communiquent pas directement : ils s'adressent les uns aux autres par l'intermédiaire d'un téléscripteur ; ils ne sauraient donc utiliser, au cours du jeu, de caractéristiques telles que l'apparence extérieure, la voix ou les performances physiques. Seul ce qui relève de l'échange linguistique est pris en compte lors du test.

Que se passera-t-il, demande Turing, si A est remplacé par une machine ? l'interrogateur C se trompera-t-il aussi souvent dans ce cas que dans l'autre ? A l'emportera-t-il aussi souvent lorsqu'il est une machine que lorsqu'il est un homme ? C'est là ce que *Computing Machinery…* s'efforce de démontrer. La machine, selon Turing, peut se comporter comme un homme placé dans les conditions du jeu - réduit à l'expression de sa pensée – c'est-à-dire comme un homme dont il est admis qu'il pense.

Computing Machinery… est souvent considéré comme l'un des textes fondateurs de l'intelligence artificielle (IA), laquelle vit officiellement le jour deux ans après la mort de Turing, en 1956[2]. De l'entreprise originelle reste aujourd'hui une thèse, dont nous empruntons l'énoncé à Jean Mosconi : « Tout comportement humain qui peut être décrit avec précision peut être simulé par un ordinateur convenablement programmé »[3]. Selon John Searle, cette thèse prend deux formes, l'une « faible », l'autre « forte » : « D'après l'IA faible, la principale valeur de l'ordinateur dans l'étude de l'esprit, c'est qu'il est pour nous un outil très puissant. Ainsi il nous permet de formuler et de tester des hypothèses de façon plus rigoureuse et plus précise. D'après l'IA forte en revanche, l'ordinateur n'est pas simplement un outil d'étude de l'esprit ; l'ordinateur convenablement programmé est véritablement un esprit, en ce sens que des ordinateurs munis des bons programmes *comprennent* et ont d'autres états cognitifs. En IA forte, l'ordinateur programmé ayant des états cognitifs, les programmes ne sont pas simplement des outils nous permettant de tester des explications ; ils sont eux-mêmes les explications. »[4]

Sous la forme de « l'IA faible », l'intelligence artificielle classique[5] est devenue une branche de l'informatique ; elle a ainsi acquis un statut : celui d'une science appliquée. Sous la forme de « l'IA forte », elle est devenue l'un des principaux points d'appui du cognitivisme, lequel, dit Pascal Engel, repose « sur l'idée que les états mentaux internes d'un organisme sont réels et peuvent être étudiés en termes de manipulations de représentations, comparables à celles que traitent les ordinateurs »[6]. Le texte de Turing,

1 Alan Mathison Turing, « Computing Machinery and Intelligence », *Mind*, 59, october 1950, p. 433-460. Publié in *Collected Works of A.M. Turing*, Londres, North-Holland, 1993, 3, *Mechanical Intelligence*. Publié en français in Jean-Yves Girard, *La machine de Turing*, trad. Patrice Blanchard, Paris, Seuil, 1995, sous le titre « Les ordinateurs et l'intelligence ».

2 Lors de la réunion de Dartmouth au cours de laquelle Newell, Shaw et Simon présentèrent un programme capable de résoudre des théorèmes simples de la logique des propositions.

3 Jean Mosconi, « Sur quelques capacités et incapacités des machines », *Bulletin de la Société Française de Philosophie*, 3, juillet-septembre 1991, p. 86.

4 John Searle, « Esprits, cerveaux et programmes », *Vues de l'esprit*, D. Hofstadter, D. Dennett, éd., Paris, InterEditions, 1987, p. 354.

5 Nous entendons par là l'intelligence artificielle « computationnelle » ou « symbolique », par opposition à l'intelligence artificielle « connexionniste » développée à partir des années 1980.

6 Pascal Engel, « La philosophie de l'esprit et les sciences cognitives », *La philosophie anglo-saxonne*, Paris, PUF, 1994, p. 533.

parce qu'il affirme que les machines peuvent « penser », a souvent été associé à cette dernière forme.

Ainsi, selon la thèse de l'IA forte, l'ordinateur *comprend*, mais qu'entend-on par « compréhension » ? Le terme renvoie d'abord à un comportement, caractérisable par différents traits, et pouvant être appréhendé comme une donnée empirique. C'est pourquoi la thèse de l'IA forte est directement concernée par le jeu de l'imitation. En premier lieu, parce que celui-ci se présente comme un test ; si l'on se met d'accord sur une description du comportement de « compréhension », et si un ordinateur, au cours d'un test adéquat, adopte un comportement correspondant à cette description, alors, la thèse de l'IA forte, selon laquelle l'ordinateur « comprend » - c'est-à-dire, en vérité, le paradigme cognitiviste - sera vérifiée. De la description d'un phénomène – le comportement de compréhension – on sera passé à la description d'un état cognitif, en l'occurrence, la description de la machine.

En second lieu, parce que Turing propose une description du phénomène de compréhension qui ne souffre guère de contestation : le jeu de l'imitation consiste, pour ses protagonistes, à avoir un échange de paroles qui peut aller de la simple conversation à la discussion, et qui porte, par hypothèse, sur l'ensemble des compétences humaines. Nul ne peut douter que l'échange de paroles, ainsi entendu, décrive bien le phénomène de la compréhension. En témoigne, du reste, la critique du jeu de l'imitation menée par Searle lui-même à l'aide de l'argument dit de la « chambre chinoise ».

Searle propose de considérer la situation suivante : un être humain qui ne comprend pas un mot de chinois est enfermé dans une pièce où il dispose de paniers dans lesquels se trouvent des symboles de la langue chinoise, et d'un livre, écrit dans sa propre langue, où sont énoncées des règles purement syntaxiques de manipulation des symboles. Des séquences de symboles chinois sont introduites dans la pièce, et les règles du livre lu par l'opérateur lui ordonnent de faire sortir de la pièce des symboles mis dans un certain ordre. Supposons que les symboles introduits dans la pièce soient des « questions » et que ceux que l'opérateur sort de la pièce soient, sans qu'il le sache, des « réponses » à ces questions. Si les règles ont été correctement rédigées, et si l'opérateur ne fait pas d'erreur en les suivant, tout se passera exactement comme si, à des questions posées par un chinois de Chine, des réponses étaient données par un chinois de Chine. Dans une telle situation, l'opérateur humain se comportera comme la machine du jeu de l'imitation. La langue dans laquelle celui-ci est joué pourrait être le chinois ; un programme P serait alors rédigé, dont l'exécution fournirait des réponses correctes en chinois à des questions posées par un examinateur chinois. Selon la thèse de l'IA forte, il découle de là qu'un ordinateur sur lequel le programme P pourrait être implémenté « comprendrait » le chinois. Or, il faudrait admettre, dans ce cas, que l'opérateur de la chambre chinoise, qui exécute lui-même le programme P, comprend le chinois, alors que, par hypothèse, il n'en connaît pas un mot[7].

La critique, ici, porte, non pas tant sur le principe du jeu que sur son résultat. Le jeu se conclut par la désignation d'un vainqueur et d'un vaincu, mais, montre Searle, si le résultat prévu par Turing – la victoire de la machine - peut, sans aucun doute, être atteint à travers un échange de paroles, il peut l'être aussi par un simple *simulacre* d'échange de paroles ; une machine peut l'emporter au jeu de l'imitation au moyen d'un comportement linguistique ne correspondant pas à la définition implicite de l'échange de paroles selon laquelle celui-ci dénote la compréhension. Nous pouvons admettre le principe sur lequel s'appuie Turing, à savoir qu'un échange de paroles allant de la simple conversation à la discussion, et impliquant l'ensemble des compétences humaines, dénote la compréhension, mais il n'est pas pour autant absolument nécessaire, pour qu'une machine sorte victorieuse du jeu de l'imitation, qu'elle soit partie prenante d'un tel échange de paroles au même sens que ses interlocuteurs. Il est possible d'imaginer une machine faisant bonne figure au jeu sans que son comportement au cours de celui-ci renvoie à un véritable équivalent de l'état cognitif de compréhension de ses adversaires. Searle, avec l'argument de la « chambre chinoise », s'efforce, en somme, de déterminer

7 « Les partisans de l'IA forte prétendent que dans cette séquence de questions-réponses, la machine ne se borne pas à simuler une capacité humaine, mais que l'on peut dire qu'elle comprend l'histoire et fournit les réponses aux questions, et que ce que font la machine et son programme explique la capacité humaine de comprendre l'histoire et de répondre à des questions la concernant. » (John Searle, *Du cerveau au savoir*, Paris, Hermann, 1985, p. 43). Pourtant, ajoute Searle, « dans une telle situation », c'est-à-dire dans la situation de l'opérateur humain placé dans la « chambre chinoise », « je vous défie d'apprendre un mot de chinois... » (*ibid*.)

le comportement minimum nécessaire à la victoire de la machine à un jeu tel que celui de l'imitation, et souligne que ce comportement est insuffisant, en lui-même, pour caractériser le phénomène de la compréhension. Pourtant, la machine victorieuse au jeu sera conforme à celles auxquelles la thèse de l'IA forte fait référence. Il y a donc des machines de l'IA forte qui peuvent l'emporter au jeu de l'imitation sans comprendre. Cela n'établit pas, à proprement parler, que la thèse soit fausse, mais au moins que le jeu de l'imitation n'est pas le bon test pour la vérifier. La question posée, à partir de là, s'agissant de *Computing Machinery…*, est de savoir s'il est légitime d'assimiler la réflexion de Turing sur le jeu de l'imitation à la thèse de l'IA forte, c'est-à-dire de chercher dans le test de Turing une confirmation anticipée de celle-ci.

Nous sommes d'emblée confrontés à un problème. Dans l'hypothèse où une lecture de *Computing Machinery…* conforme à la théorie de l'IA forte fournirait une interprétation fidèle de la réflexion de Turing, la question examinée par ce dernier : « Les machines peuvent-elles penser ? » ne serait-elle pas avantageusement remplacée par cette autre : « Les machines peuvent-elles comprendre ? » ? Si *Computing Machinery…* devait être considéré comme une première formulation de la thèse de l'IA forte, non seulement il n'y aurait pas ici trahison – puisque la thèse consiste à affirmer que la machine comprend – mais, en outre, la question gagnerait en précision. En l'absence de toute élaboration conceptuelle, le terme « penser », dans la question posée par Turing, reste en effet redoutablement vague. N'est-ce pas, d'ailleurs, précisément pour cette raison que Turing propose de remplacer la question : « Les machines peuvent-elles penser ? » par une « expérience » imaginaire - le jeu de l'imitation[8] - qui lui permet d'éviter tout à la fois l'imprécision de la question et les difficultés conceptuelles liées au terme « penser » ? Sous cet angle, l'interprétation de sa démarche conforme à la théorie de l'IA forte consiste manifestement à considérer que le remplacement de la question : « Les machines peuvent-elles penser ? » par le jeu de l'imitation équivaut à traduire la question initiale sous la forme de questions telles que : « Les machines peuvent-elles comprendre ? », ou encore : « Les machines peuvent-elles être intelligentes ? », questions qui peuvent être traitées comme des problèmes de psychologie cognitive. Or, il y a, nous semble-t-il, inadéquation entre ces questions mêmes et la structure spécifique du jeu de l'imitation.

On notera, en effet, que s'il s'agissait de répondre à la question « Les machines peuvent-elles comprendre ? », plutôt qu'à la question : « les machines peuvent-elles penser ? », le jeu de l'imitation ne serait probablement pas l'expérience la plus significative ni la plus pertinente : à quoi bon demander à un examinateur de distinguer lequel, de deux interlocuteurs, est un homme et lequel est une machine ? Ne suffirait-il pas de vérifier l'aptitude d'une machine à avoir avec un être humain quelconque un échange de paroles pouvant aller de la simple conversation à la discussion ? C'est du reste de cette manière que Searle lui-même, visant la thèse de l'IA forte à travers le jeu de l'imitation, traduit celui-ci dans son expérience de la chambre chinoise : il n'y s'agit plus de distinguer un homme d'une machine, mais d'imaginer une situation d'échange de paroles entre deux interlocuteurs humains dont l'un se comporte comme une machine.

Dans le jeu de l'imitation, tel que l'expose Turing, un homme, lui-même considéré comme pensant, doit distinguer un homme d'une machine et, pour cela, *reconnaître la pensée*. Compte tenu, en effet, des conditions dans lesquelles le jeu doit se dérouler, à savoir une séparation « assez nette entre les capacités physiques et intellectuelles de l'homme »[9], le jeu ne consiste pas, pour l'examinateur, à dire : « j'ai reconnu un homme, c'est-à-dire un être pensant, donc l'entité que j'ai reconnue pense », mais, à l'inverse : « je reconnais un discours, c'est-à-dire de la pensée, donc l'entité que je reconnais est un homme ». Par là même, on voit bien que la réflexion de Turing n'évite guère, en remplaçant la question de la « pensée » des machines par son test, les difficultés proprement philosophiques du terme « penser » ; Turing affronte, à travers le jeu de l'imitation lui-même, une question que la théorie de l'IA forte n'entend jamais poser : comment la pensée pourrait-elle être déduite de la non-pensée ? La pensée ne peut être reconnue, tout comme, du reste, l'absence de pensée, que de l'intérieur de la pensée ; la pensée est toujours déjà-là, présente à elle-même. De sorte qu'il n'est pas exclu que ce soit en référence à ce que le terme « penser » conserve d'opacité conceptuelle que

8 « Je crois que la question originale "Les machines peuvent-elles penser ?" a trop peu de sens pour mériter une discussion. », « Les ordinateurs et l'intelligence », *op. cit.*, p. 148.
9 *Ibid.*, p. 136.

Turing, plutôt que de discuter ce terme à travers la question « les machines peuvent-elles penser ? », décide de remplacer la question elle-même par l'« expérience » du jeu de l'imitation. Tout se passe comme si celui-ci, dans son économie singulière, et en tant qu'il se substitue à la question de la « pensée » des machines, renvoyait le terme « penser » à un usage qui comporte en lui-même le fait de savoir que le terme « penser » ne peut être énoncé sans présupposer le penser du penser, c'est-à-dire sans faire du penser un principe d'intelligibilité, sans présupposer la transparence du penser à lui-même, ou encore sans présupposer que le penser est son propre principe d'effectivité. Que Turing l'ait consciemment voulu ou non, le paradoxe du penser, qui tient à ce que son opacité est celle de l'intelligibilité même, est présent au cœur du jeu de l'imitation. En d'autres termes, le problème auquel renvoie le jeu de l'imitation ne peut pas être réduit à celui de la compréhension, au sens cognitif du terme.

Il reste que l'idée selon laquelle le jeu de l'imitation serait une première expression de la thèse de l'IA forte peut s'appuyer sur une série d'arguments solides : la machine de l'IA forte n'est-elle pas celle-là même à laquelle Turing entend faire jouer le jeu de l'imitation ? La thèse de Turing n'est-elle pas que cette machine, qui peut réussir le test du jeu, « comprend » ? Enfin, la méthode de l'IA forte – liée au type de machine considéré – n'est-elle pas celle qu'envisageait Turing pour construire une machine capable de l'emporter au jeu ? Nous devons donc examiner ces trois questions.

II

La première ne présente pas de difficulté. Dans le cas du jeu de l'imitation, la machine envisagée par Turing est celle à laquelle il a donné son nom et qu'il a décrite dans l'article sur les « nombres calculables » par lequel il s'est fait connaître en 1937[10] : une « machine de Turing », machine à « états discrets », pouvant calculer n'importe quelle fonction calculable par un calculateur humain. Turing a établi qu'une machine de Turing pouvait être conçue comme une « machine universelle » capable de simuler l'action de n'importe quelle « machine de Turing ». Or, la notion de machine universelle définit précisément l'ordinateur tel que nous le connaissons aujourd'hui, et tel que le conçoit l'IA forte. La machine du jeu de l'imitation et celle de l'IA forte sont donc une seule et même machine.

Qu'en est-il de la thèse énoncée par Turing ? Celui-ci affirme-t-il que la machine universelle qui l'emporte au jeu de l'imitation *comprend* ? Turing note que l'on ne peut répondre à la question « Les machines peuvent-elles penser ? » en ayant recours à une définition des termes « machine » et « penser » qui serait établie à partir de « sondages d'opinion ». Sans doute ceux-ci montreraient-ils que, pour la majorité des gens interrogés, les machines ne peuvent pas penser, mais en l'occurrence la méthode serait « absurde »[11]. Le recours au « sondage d'opinion » ne ferait qu'attester *l'usage*, à un certain moment, des termes « machine » et « penser » et fournirait seulement, à la question « Les machines peuvent-elles penser ? », la réponse susceptible d'être faite à ce moment. Dire cela n'est pas condamner le recours à l'usage, mais prendre en compte le fait que ce recours impose de déterminer s'il ne peut exister une situation impliquant un usage des termes « machine » et « penser » au regard duquel la réponse à la question serait positive. Mettre en présence d'une telle situation est précisément ce que vise à accomplir le jeu de l'imitation. C'est pourquoi Turing s'aventure à prédire que, cinquante ans après la publication de son article, non seulement les progrès réalisés en matière de conception de « machines universelles » seront suffisants pour que cette situation – la victoire d'une machine au jeu – soit envisageable en pratique, mais encore que l'usage,

10 Alan Mathison Turing, « On Computable Numbers with an Application to the *Entscheidungsproblem* », *Proceedings of the London Mathematical Society*, vol 42, 1937. Publié in *Collected Works of A. M. Turing*, *op. cit.*, 2, *Mathematical Logic*. Publié en français in Jean-Yves Girard, *La machine de Turing*, trad. Julien Basch, *op. cit.*, sous le titre « Théorie des nombres calculables, suivie d'une application au problème de la décision ».

11 « Il faudrait commencer par définir le sens des termes "machine" et "penser". Les définitions peuvent être conçues de manière à refléter l'utilisation normale des mots, mais cette attitude est dangereuse. Si on doit trouver la signification des mots "machine" et "penser" en examinant comment ils sont communément utilisés, il est difficile d'échapper à la conclusion que la signification de la question "Les machines peuvent-elles penser ?" et la réponse à cette question doivent être recherchées dans une étude statistique telle que le sondage d'opinion. Mais cela est absurde. », « Les ordinateurs et l'intelligence », *op. cit.*, p. 135.

en ce qui concerne le terme « machine », aura suffisamment évolué pour que l'affirmation de la « pensée » des machines ne choque plus le sens commun.

En quoi, néanmoins, le recours à l'usage peut-il être une bonne méthode ? N'est-ce pas en ce sens que l'usage du terme « penser » implique la compréhension ? C'est en fonction d'un usage où le terme « penser » signifie notamment « comprendre » qu'une enquête d'opinion montrerait, en 1950, que la réponse de la majorité des gens interrogés à la question « Les machines peuvent-elles penser ? » serait négative. Quant à la modification de l'usage qui doit conduire à revenir sur cette négation du « penser » de la machine, elle ne porte pas sur l'usage du terme « penser », mais seulement sur celui du terme « machine » ; si l'on dit, en l'an 2000, après une victoire de la machine au jeu de l'imitation, qu'une machine peut « penser », ce sera toujours en référence à un usage du terme « penser » incluant la « compréhension ».

On ne pourra pas en conclure, pourtant, que la thèse de Turing est identique à celle de l'IA forte : en montrant qu'une machine peut l'emporter au jeu de l'imitation, Turing n'affirme pas que cette machine « pense », et donc « comprend », mais plutôt que la victoire d'une machine au jeu de l'imitation rend possible de *dire* qu'elle « pense », en d'autres termes, que la victoire de la machine détermine un usage du terme « machine » dans le cadre duquel l'affirmation qu'elle pense, et donc comprend, prend sens. C'est bien, d'ailleurs, ce qui ressort de la manière même dont, selon Turing, la machine peut faire bonne figure au jeu : la machine l'emporte sur ses adversaires si l'examinateur est conduit à se comporter *comme si*, à ses yeux, elle comprenait.

Turing, en effet, pour démontrer qu'une machine peut l'emporter au jeu de l'imitation, examine deux catégories de questions : celles qui relèvent de la logique mathématique, à travers ce qu'il nomme « l'objection mathématique », et toutes les autres, sous l'angle de deux problèmes principaux : celui de la conscience et celui de « l'informalité du comportement ».

S'agissant de l'objection mathématique, Turing fait référence, on le sait, aux théorèmes de limitation - notamment au théorème de Gödel – selon lesquels, résume-t-il, « dans tout système logique suffisamment puissant on peut formuler des affirmations qui ne peuvent ni être prouvées ni être réfutées à l'intérieur du système, à moins que le système lui-même ne soit inconsistant » [12]. Ainsi, nous savons qu'il existe, dans le cadre du jeu de l'imitation, certaines questions auxquelles une machine « à états discrets », fût-elle une machine universelle, ne pourra jamais répondre correctement, au regard d'un examinateur tel que celui du jeu. Ce ne sont pas l'ignorance ou l'incompétence de la machine qui sont visées ici. Nous pouvons supposer, bien entendu, qu'une machine aura du mal à répondre à une question lui demandant ce qu'elle pense de Picasso, pourtant, il n'est pas en principe exclu qu'elle puisse le faire : rien ne s'oppose, en théorie, à ce qu'une machine soit directement programmée pour simuler des réponses sensées, aux yeux d'un examinateur humain quelconque, à propos de Picasso. L'objection mathématique porte sur un tout autre type de questions : celles à propos desquelles se pose le problème logique de la décision, par exemple celles auxquelles on doit répondre par « oui » ou par « non »[13]. Certaines de ces questions sont décidables : il existe une procédure effective – un algorithme – permettant de choisir entre le « oui » et le « non » ; ainsi de la question : « *n* est-il un nombre premier ? ». D'autres ne sont pas décidables : il n'existe pas de procédure effective permettant de trancher entre le « oui » et le « non ». Tel est le cas, rappelle Turing, de la question : « considérez la machine spécifiée comme suit… Est-ce que cette machine répondra toujours "oui" à toute question ? » [14]. Turing lui-même a établi qu'il est logiquement impossible qu'une machine conforme à la définition de la machine universelle puisse fournir à cette question une réponse satisfaisante : en tant que machine, elle restera bloquée dans l'état correspondant pour elle à la question, sans jamais pouvoir en sortir.

A la racine de la limitation logique des machines se trouve le fait, établi à partir du théorème de Gödel, que, pour toute machine, une certaine formule peut être construite à l'aide des éléments que la machine manipule alors que cette formule ne peut être calculée par elle ; la formule considérée ne peut être calculée que par un système plus

12 « Les ordinateurs et l'intelligence », *op. cit.*, p.151.

13 « Nous supposons bien sûr pour le moment des questions appelant une réponse en "oui" ou en "non", plutôt que des questions telles que : "Que pensez-vous de Picasso ?" », *ibid.*, p. 152.

14 « Nous savons que les machines doivent échouer dans des questions du type : "Considérez la machine spécifiée comme suit… Cette machine répondra-t-elle 'oui' à n'importe quelle question ?'". », *ibid.*

puissant. N'est-ce pas à dire que l'esprit humain, précisément capable, quant à lui, d'élaborer le système plus puissant permettant de calculer cette formule, est supérieur à la machine ? Bien plus, et c'est l'argument auquel s'attaque Turing, n'est-ce pas à dire que les théorèmes de limitation prouvent une impuissance de la machine qui n'affecte pas l'esprit humain ?

Cependant, dès lors que la formule devant laquelle s'arrête telle machine peut être *calculée* dans le cadre d'un système plus puissant, ne peut-elle être calculée par une machine correspondant à ce système plus puissant ? Il y aura, certes, pour cette seconde machine comme pour la première, une formule qui l'arrêtera, qu'un esprit humain sera capable de calculer, mais une troisième machine, représentant le système plus puissant à partir duquel la formule qui arrête la seconde est calculée, pourra à son tour être construite, et ainsi de suite[15].

Par ailleurs, si les théorèmes de limitation prouvent bien une limitation du pouvoir de la machine, ils ne prouvent nullement qu'il n'y a pas une limitation analogue pour l'esprit humain[16]. Les hommes se trompent eux-mêmes très souvent, note Turing[17]. Sans doute, dans le cadre du jeu de l'imitation, l'examinateur pourra, à un moment ou un autre du déroulement de l'échange avec la machine, poser à celle-ci une question à laquelle elle ne saura pas répondre ; cependant, il serait tout à fait possible de construire une machine, qui, après un certain délai, ou un certain nombre de sollicitations de la part de l'examinateur humain du jeu, répondrait : « je ne sais pas ». En d'autres termes, nous pouvons imaginer, dans le cadre du jeu, une machine qui, lorsqu'elle est en difficulté, donne une réponse erronée, ou bien « dise » : « je ne sais pas ». Or, dès lors qu'un homme peut, lui aussi, se tromper très souvent ou avouer son ignorance, comment cela permettrait-il à l'examinateur A de distinguer la machine C de B, son partenaire humain dans le jeu ?[18]

En outre, avant de s'appliquer aux compétences logiques des machines, les théorèmes de limitation s'appliquent à celles des hommes eux-mêmes, de sorte que, puisqu'une machine peut répondre à la question qui en arrête une autre dans les mêmes termes que le ferait l'esprit humain, et puisque la puissance logique de celui-ci est elle-même limitée, il n'y a pas de raison *logique*, ou mathématique, de supposer qu'il ne puisse pas toujours y avoir une machine capable de faire aussi bien que l'esprit humain ; il n'y a aucune raison logique d'affirmer qu'il existe, dans la série des questions « critiques » qui arrêtent chaque machine, un degré auquel l'esprit humain ne pourra pas être rattrapé par une machine.

Cependant, le jeu de l'imitation ne sollicite pas seulement les capacités logiques ou mathématiques de ses protagonistes, mais l'ensemble des compétences d'un être humain quelconque, de sorte qu'en montrant qu'il n'y a pas d'objection mathématique à une victoire de la machine au jeu, on n'a pas encore établi que cette victoire est possible : il faut en outre montrer que la machine n'est pas vouée à l'échec lorsqu'elle est soumise à d'autres épreuves que des tests mathématiques, épreuves qu'un homme, quant à lui, réussirait. La principale question ainsi abordée par Turing est, de manière classique, celle de la conscience ; le penser, en l'homme, est inséparable de la conscience, c'est pourquoi, comme le dit le professeur Jefferson, cité par Turing, « Nous ne pourrons pas accepter l'idée que la machine égale le cerveau jusqu'à ce qu'une machine puisse écrire un sonnet ou composer un concerto à partir de pensées ou d'émotions ressenties et non pas en choisissant des symboles au hasard, et non seulement l'écrire, mais savoir qu'elle l'a écrit »[19].

Qu'arrivera-t-il si, au cours du jeu de l'imitation, l'examinateur C demande à son interlocuteur A – la machine - de composer un sonnet ? Admettons qu'une machine

15 Turing abordait la question sous son aspect formel dans la la thèse qu'il avait soutenue sous la direction d'Alonzo Church en 1938 à Princeton : A. M. Turing, « Systems of Logic Based on Ordinals », *Proceedings of the London Mathematical Society*, 2, 45, 1939.

16 « ... bien qu'il soit établi qu'il y a des limites à la puissance de n'importe quelle machine, il a seulement été affirmé, sans aucune sorte de preuve, que de telles limites ne s'appliquaient pas à l'esprit humain. ». « Les ordinateurs et l'intelligence », *op. cit.*, p. 152.

17 « Nous donnons nous-mêmes trop souvent des réponses fausses à des questions pour que nous ayons le droit de nous réjouir d'une telle preuve de la faillibilité des machines. ». *Ibid.*, p.153.

18 « Chaque fois que l'on pose à l'une de ces machines la question cruciale appropriée et qu'elle donne une réponse définie, nous savons que cette réponse est forcément fausse, ce qui nous procure un certain sentiment de supériorité. Ce sentiment est-il illusoire ? Il est sans aucun doute tout à fait sincère, mais je ne pense pas qu'il faille y attacher trop d'importance. », *ibid.*

19 *Ibid.*

puisse être programmée pour composer un sonnet – après tout, le sonnet renvoie à une forme qui peut être décrite comme un ensemble de règles - ; cela ne suffira pas à prouver qu'elle « pense », c'est-à-dire, en l'occurrence, que le sonnet qu'elle propose soit bien l'expression d'émotions réfléchies dans la forme poétique. Cependant, il existe, selon Turing, un moyen de vérifier que la machine, non seulement peut composer un sonnet, mais encore sait de quoi elle parle, et sait qu'elle a écrit un sonnet : ce moyen est celui qu'utilise en général un examinateur pour vérifier que le candidat à un examen a réellement compris les questions qui lui sont posées et les réponses qu'il y fait : que se passera-t-il si l'examinateur C demande à son interlocuteur A de *commenter* le sonnet qu'il vient de composer ? Si C est le professeur Jefferson, il sera certainement satisfait, remarque Turing, pour peu que A fournisse des réponses sensées – sans être nécessairement exactes - à des questions telles que : « pourquoi avez-vous utilisé tel terme ? », ou : « qu'avez-vous voulu dire par là ? ». Il s'agit donc de montrer qu'une telle performance n'est pas, dans son principe, hors de portée de la machine. C'est à quoi tend l'étude que fait Turing de problèmes tels que celui de la conscience réflexive – une machine peut-elle être « l'objet de ses propres pensées » ? – ou celui de ce qu'on peut appeler la capacité d' « invention » – une machine peut-elle surprendre son interlocuteur humain de la même façon que le ferait un individu humain ? Il ressort précisément de cette étude que l'examinateur du jeu n'aura, selon Turing, aucun moyen de distinguer, s'agissant de la conscience réflexive ou de la capacité d'invention, le comportement de la machine A de celui de son partenaire humain B. On peut dire d'une machine universelle, dont le propre est de pouvoir simuler d'autres machines, et qui peut être considérée, à chaque moment du jeu, comme le couple formé par une machine simulée et une machine qui simule celle-ci, qu'elle se prend elle-même pour objet. Il est par ailleurs facile d'établir qu'un observateur humain peut être surpris par le comportement d'une machine de complexité moyenne[20] ; or, montre Turing, cet observateur ne sera certainement pas en mesure, dans un contexte tel que celui du jeu de l'imitation, de distinguer l'effet de surprise provoqué par une machine de celui qu'il peut éprouver face à un interlocuteur humain inventif.

Turing conclut sa réflexion sur ce point en abordant le problème plus général de « l'informalité du comportement humain » : on ne saurait produire un ensemble complet de règles qui permettent de prévoir ce que fera un homme dans toute circonstance. Turing admet cette proposition, mais refuse l'argument qui en est tiré, à savoir : « Si chaque homme disposait d'un ensemble défini de règles de conduite d'après lesquelles il organiserait sa vie, il ne serait pas supérieur à la machine ; mais de telles règles n'existent pas ; ainsi les hommes ne peuvent pas être des machines »[21].

La faiblesse de l'argument tient, pour Turing, dans le moyen terme : « mais de telles règles n'existent pas ». N'y a–t-il pas, ici, demande-t-il, confusion entre « règles de conduite » et « lois du comportement » ? Par « règles de conduite », Turing entend, précise-t-il, des préceptes sur lesquels on peut agir et dont on peut être conscient. Quant aux « lois du comportement », le lecteur doit y voir un équivalent des lois de la nature, c'est-à-dire des déterminations dont nous ne sommes pas conscients au moment où nous les suivons, et sur lesquels nous ne pouvons agir.

L'argument peut donc être reformulé en substituant l'expression « lois du comportement » à celle de « règles de conduites ». Nous pourrons alors accorder le premier terme : « si chaque homme disposait d'un ensemble défini de lois du comportement qui règlent sa vie, il ne serait pas supérieur à la machine ». Cependant, nous ne pourrons pas accorder aussi facilement le deuxième : « mais de telles lois n'existent pas ». De même, en effet, que ce n'est pas parce que nous n'avons pas encore trouvé toutes les lois de la nature que nous sommes en droit de conclure que de telles lois n'existent pas, de même, ce n'est pas parce que nous n'avons pas trouvé les « lois du comportement » que nous avons le droit d'affirmer que ces lois n'existent pas. Dans le cadre du jeu, cela est d'autant plus déterminant que nous ne pouvons pas espérer connaître les « lois du comportement » de la machine, comme le montre Turing, à l'aide de la seule observation[22].

---

20 « Les machines me prennent très fréquemment par surprise. », *ibid.*, p. 161.

21 *Ibid*., p. 163.

22 « J'ai introduit dans l'ordinateur de Manchester un petit programme utilisant seulement mille unités de stockage, par lequel la machine, lorsqu'on lui fournit un nombre de seize chiffres, répond par un autre nombre en deux secondes. Je défie quiconque d'en apprendre assez au sujet du programme à partir de ces réponses

Turing, en somme, dans *Computing Machinery*…, établit l'égalité de la machine et de l'esprit humain sur le plan logique, la possibilité pour la machine de simuler la conscience, l'impossibilité, enfin, pour l'examinateur du jeu de prévoir le comportement de la machine mieux que celui d'un homme. De là résulte, d'une part, que, sur le plan du calcul proprement dit, tout ce que fait un homme pourra être effectué également par une machine, d'autre part, que ce qui ne rentre pas dans le champ du calcul proprement dit n'est pas susceptible d'un critère de discrimination sans équivoque, et, par conséquent, que l'examinateur C ne dispose *a priori* d'aucun moyen sûr de faire le départ entre le comportement de A – la machine – et celui de B – l'autre humain du jeu. C peut être ainsi conduit à agir, au cours du jeu, *comme si* A était, pour lui, un être humain.

La thèse qui découle de l'hypothèse de la victoire d'une machine au jeu de l'imitation n'est donc pas exactement celle de l'IA forte ; strictement parlant, dans le cadre de l'hypothèse qu'il présente, Turing n'affirme pas tant la pensée effective de la machine que la possibilité d'une situation impliquant un usage du terme « machine » au regard duquel nous avons le droit d'affirmer cette pensée comme nous le faisons à l'égard d'un interlocuteur humain.

Qu'en est-il, cependant, de la méthode de Turing comparée à celle de l'IA forte en ce qui concerne la réalisation effective d'une machine susceptible de l'emporter au jeu de l'imitation ?

III

Le principe d'une lecture de la démarche de Turing conforme à la thèse de l'IA forte sera celui-là même sur lequel repose l'analyse de l'objection mathématique, c'est-à-dire la définition de la machine universelle, machine de Turing qui peut simuler l'action de toute autre machine de Turing. Dans le cas de l'objection mathématique, la machine arrêtée par sa question critique peut être relayée par une autre qui la simule tout en étant capable, parce qu'elle dispose d'un axiome supplémentaire, de traiter la question critique ; cette machine rencontre elle-même sa propre question critique, mais elle peut être relayée par une autre, et ainsi de suite. On figure par là, à l'aide d'une série ascendante de machines, une machine universelle dont la puissance logique a la même étendue que celle de l'esprit humain.

Par ailleurs, dans les domaines non mathématiques, pour chaque situation concrète possible dans le cadre du jeu de l'imitation, une machine peut être programmée pour simuler le comportement d'un individu humain de telle manière que l'examinateur humain du jeu n'ait aucun moyen bien défini de faire le départ entre ce qu'il attend d'un humain et ce que fait la machine. Par là, le principe mis en avant pour vaincre l'objection mathématique peut être étendu : en vertu de la définition même de la machine universelle, rien ne s'oppose, en théorie, à ce qu'une machine puisse simuler un nombre suffisant de machines capables de l'emporter à ce que nous pourrions appeler des « tests partiels » de Turing, pour sortir victorieuse du jeu de l'imitation proprement dit, qui, on le sait, sollicite l'ensemble des compétences humaines. Quel que soit le domaine que l'examinateur choisira d'aborder, il est théoriquement possible, selon Turing, d'imaginer une machine que l'examinateur, dans la situation du jeu, ne pourra distinguer d'un individu humain ; et il est, enfin, théoriquement possible d'imaginer une machine universelle simulant chacune des machines l'emportant à des « tests partiels ».

Pour l'IA forte, construire une machine « qui comprend » consiste à élaborer le code - le texte du programme - qui la définit. La définition même de la machine universelle conduit à envisager l'emboîtement des machines l'emportant à des « tests partiels de Turing » selon cette méthode : chaque machine sera définie par son programme, exprimable dans un certain langage, et la machine universelle qui simule chacune d'elles sera elle-même définie comme le programme codant l'emboîtement des programmes définissant ces machines spécifiques. Toute la question est de savoir comment cette machine « emboîtante » sera elle-même spécifiée.

C'est ici qu'intervient la seconde hypothèse examinée par Turing, dans la dernière partie de *Computing Machinery*[23], celle des « machines qui apprennent ». Or, cette

pour être capable de prédire la réponse pour des valeurs non encore utilisées. », *ibid.*, p. 164.

23 Il s'agit de la section 7, qui correspond à la troisième grande partie du texte.

seconde hypothèse ouvre une perspective dans laquelle, paradoxalement, la thèse de Turing se rapproche de celle de l'IA forte – la machine semble « comprendre » - alors même que sa méthode se distingue nettement de celle de l'IA forte.

Pour mesurer la portée de l'hypothèse des « machines qui apprennent », nous devons tout d'abord examiner la critique généralement faite au jeu de l'imitation dans le cadre de la réfutation de la thèse de l'IA forte.

Si l'on conserve l'idée que la machine victorieuse au jeu de l'imitation correspond à une série ascendante de machines, alors, le point important sera, ici, que la machine victorieuse ne passe pas *par elle-même* d'un moment de la série à l'autre. Il faut qu'à chaque étape, des hommes *construisent* la machine qui, tout en faisant ce que fait la précédente, effectue aussi quelque chose que celle-ci ne sait pas faire. Pour reprendre la terminologie utilisée par Turing dans sa thèse américaine sur les « logiques ordinales », la machine doit avoir recours à un « oracle »[24]. Dans le cas du jeu de l'imitation, cet oracle sera l'homme, capable, quant à lui, de calculer la formule inaccessible à telle machine, et de construire la machine plus puissante, ou aux compétences de simulation plus étendues, correspondant à ce calcul. Par là même, il est clair que ce qui *comprend*, dans le cadre de la victoire de A au jeu de l'imitation, ce n'est pas la machine représentant la série des machines successives, ou correspondant à l'emboîtement des machines l'emportant à des « tests partiels de Turing », mais « l'oracle », c'est-à-dire l'homme – ou l'équipe d'hommes – qui a construit les machines successives[25].

C'est là, d'une certaine façon, ce qui ressort de la critique du jeu menée par Searle à l'aide de son argument de la « chambre chinoise », dans le cadre de sa discussion de la thèse de l'IA forte : la « machine » de la chambre chinoise fournit à ses interlocuteurs des réponses qu'ils comprennent alors qu'elle ne les comprend pas elle-même ; le sens de ses réponses est tout entier dans les règles qui lui sont fournies par ses programmeurs.

Or, cette critique tombe, dans son principe, dès lors que l'hypothèse de la victoire possible d'une machine au jeu de l'imitation est interprétée à partir de la seconde hypothèse défendue par Turing, celle des « machines qui apprennent ».

Turing introduit sa réflexion sur les « machines qui apprennent » en s'efforçant de caractériser le penser chez l'homme à l'aide de la notion de « surcriticalité », empruntée à la physique. L'esprit humain peut, selon lui, être comparé à une pile atomique : tant que la masse d'une pile atomique n'a pas atteint un certain seuil, elle reste « sous-critique » et l'entrée d'un neutron dans la pile provoque une perturbation qui cesse d'elle-même au bout d'un certain temps ; lorsque la masse a atteint le seuil requis, la pile devient « sur-critique » et l'entrée d'un neutron provoque une perturbation qui continue à se développer jusqu'à destruction de la pile. Turing suggère quelque chose d'analogue en ce qui concerne le fonctionnement de l'esprit humain : tant qu'il reste « sous-critique », l'injection en lui d'une idée donne lieu, « en moyenne, à l'apparition de moins d'une idée en réponse »[26] ; lorsqu'il devient « sur-critique », l'injection d'une idée « pourra donner lieu à l'apparition de toute une "théorie" constituée d'idées secondaires, tertiaires ou encore plus éloignées »[27]. D'où la question posée alors par Turing : « Peut-on rendre une machine sur-critique ? »[28]. S'il est vrai, en d'autres termes, que penser, pour l'esprit humain, implique la possibilité de devenir « sur-critique », alors, la machine qui l'emporte au jeu de l'imitation est-elle une machine potentiellement sur-critique ? Bref, tout se passe comme si Turing entendait bien vérifier, ici, que la victoire de la machine au jeu suppose, non seulement une *simulation* du comportement interprété comme du

---

24 Turing relevait que le théorème de Gödel, qui établit que tout système logique est incomplet, montrait par là même comment « from a system L of logic a more complete system L' may be obtained. By repeating the process we get a sequence L, L1=L', L2=L1', ... each more complete than the preceding ». Ainsi, « a logic Lω may then be constructed in wich the provable theorems are the totality of theorems provable with the help of the logics L, L1, L2, ... », et « We may then form L2ω related to Lω in the same way as Lω was related to L. » (« Systems of Logic Based on Ordinals », *op. cit.*, p. 161). Traduit dans les termes de l'article sur les « nombres calculables », le système Lω consistait en une suite de « machines de Turing » représentées chacune par un nombre ordinal. Pour représenter le passage d'une machine à une autre, Turing imaginait le recours à un « oracle » : chaque machine devait disposer d'un état dans lequel elle s'en remettait à cet « oracle » pour connaître la configuration de la machine suivante.

25 Ce problème a été soulevé par J.R. Lucas. Voir : J. R. Lucas, « L'esprit humain, la machine et Gödel », Alan Ross Anderson, Gérard Guièze (dir.), *Pensée et machine*, Paris, Champ Vallon, 1983.

26 *Ibid.*, p. 167.

27 *Ibid.*

28 *Ibid.*

« penser » par l'examinateur du jeu, mais le « penser » en tant que tel[29].

La comparaison avec la pile atomique suggère que la machine sur-critique est celle dans laquelle, à partir d'un certain seuil de complexité, l'introduction d'un élément nouveau provoque une explosion combinatoire. D'où il ressort, comme le fait remarquer Turing, que le degré de complexité atteint par la machine ne permet plus aux constructeurs mêmes de la machine, et *a fortiori* à un observateur extérieur comme l'examinateur du jeu, de prévoir le comportement de celle-ci. Bref, l'analogie de la machine sur-critique avec l'esprit humain met en jeu le niveau de complexité atteint par l'un et l'autre. Or, si le niveau de complexité de l'esprit humain interdit que le comportement humain soit strictement prévu, ce comportement, cependant, peut, par l'éducation, être orienté. L'éducation repose sur la faculté d'apprendre ; la question de la « surcriticalité » de la machine peut en conséquence être ramenée à celle de l'apprentissage. Si une machine peut être sur-critique, elle doit pouvoir apprendre, c'est-à-dire modifier ses propres programmes, bref, agir sur elle-même. Enfin, si une machine peut apprendre, la meilleure méthode pour construire celle qui l'emportera au jeu de l'imitation, lequel, rappelons-le, sollicite l'ensemble des compétences humaines, sera certainement de « l'éduquer ».

Turing examine l'hypothèse des « machines qui apprennent » en vérifiant que l'idée d'un processus d'apprentissage mettant en jeu une « surcriticalité », qui renvoie elle-même, sans autre précision, et donc sans autre restriction, au « penser » humain, n'entre pas en contradiction avec la définition de la machine universelle. Il s'efforce de le montrer en exploitant la dimension particulière de celle-ci selon laquelle elle peut être décrite comme une « machine inorganisée » (« non-déterministe »)[30]. Il est possible, selon Turing, par un processus d'essais et d'erreurs commandé par une simulation, dans les termes de la machine, du système « punitions-récompenses » utilisé dans l'apprentissage et dans l'éducation des hommes, de transformer une machine inorganisée en une machine organisée, machine universelle, qui, par le même processus, pourra « apprendre ».

Dans le rapport intitulé *Intelligent Machinery*, destiné au *National Physical Laboratory*, et rédigé quelques mois avant *Computing Machinery...*[31], Turing expose la manière dont il conçoit une « machine qui apprend ». Trois éléments déterminent l'action d'une telle machine : en premier lieu, sa description logique sous la forme d'une table indiquant ses états et les règles associées à ceux-ci – ce que Turing appelle dans *The Computable Numbers...*, la « m-configuration » de la machine, et qui peut être assimilé au « code » du programme. Turing, dans *Intelligent Machinery*, parle du « caractère » de la machine. En second lieu, la configuration matérielle qui « implémente » cette description logique, que Turing appelle, ici, la « situation » de la machine[32]. Enfin, le signal d'entrée par lequel le milieu extérieur entre en contact avec la machine[33]. A tel signal d'entrée la machine répondra par une action que détermine son code et l'implémentation matérielle de celui-ci. Si l'action est considérée comme inadéquate, la machine recevra un « stimulus de peine » : elle sera « punie ». La punition consistera en un signal lui ordonnant de changer son code, selon une procédure aléatoire[34], et, par suite,

29 Sous un certain angle, la notion turingienne de « surcriticalité » rejoint les réflexions sur les automates très complexes émises par Von Neumann à peu près au même moment – en décembre 1949 – dans une série de conférences devant l'université de l'Illinois (John Von Neumann, *Theory of self-reproducing automata*, Londres, University of Illinois Press, 1966). Von Neumann notait que plus un système est complexe, plus grande est la probabilité d'erreurs de fonctionnement de ce système. Il montrait alors qu'en deçà d'un certain seuil, la complexité est « dégénérative » car les erreurs de fonctionnement sont bloquantes, mais qu'au-delà de ce seuil, le système devient assez complexe pour pouvoir corriger lui-même les erreurs de fonctionnement. Ce seuil est ce qui à ses yeux distingue les automates naturels des automates artificiels : une fois franchi ce seuil, l'automate devient capable d'agir sur lui-même.

30 Turing mentionnait, dès l'article sur les « nombres calculables », la possibilité de concevoir une machine de Turing non déterministe : « Dans certains cas, on peut avoir besoin d'une machine à choix (*c*-machine), dont le comportement ne dépend pas entièrement de sa configuration... », « Théorie des nombres calculables... », *op. cit.*, p. 52. A certaines étapes du processus suivi par une telle machine, deux choix, au moins, sont possibles, et la machine est conçue de manière à ce que l'un ou l'autre choix soit déterminé par une procédure faisant intervenir un opérateur extérieur ou le hasard - un équivalent mécanique du coup de dé. Dans le rapport intitulé *Intelligent Machinery*, qu'il avait rédigé en 1947 à l'intention du *National Physical Laboratory*, Turing décrit ainsi une machine « aléatoire » (« random machine »), qu'il appelle également une « machine inorganisée ».

31 « Intelligent Machinery », in *Collected Works of A.M. Turing*, 2, « Mechanical Intelligence », *opus cit.*

32 « the configurations of the machine are described by two expressions, which we may call the character-expression and the situation-expression », « Intelligent Machinery », p. 121.

33 « The character and situation at any moment, together with the input signals, determine the character and situation at the next moment. », *ibid.*

34 Par exemple à l'aide d'une fonction jouant le rôle de coup de dé électronique.

l'implémentation matérielle correspondante. Si, au contraire l'action est considérée comme adéquate, la machine recevra un « stimulus de plaisir » : elle sera « récompensée ». La récompense consistera en un signal ordonnant à la machine de fixer le code « gagnant » et l'implémentation matérielle correspondante. Sur une durée très longue, la machine tendra à rencontrer des situations de plaisir et à fixer les configurations correspondantes. On voit bien, alors, qu'une situation complexe, du type de celles qui peuvent être impliquées dans le jeu de l'imitation, peut être traduite sous la forme d'un ensemble de signaux d'entrée, auxquels sont associés un code et une implémentation matérielle. Du point de vue de cette situation complexe, la probabilité des réponses « fausses » de la machine, celles engendrant une « punition », tendra à décroître. On peut formuler l'idée que, face à une telle situation complexe, la machine aura « appris » lorsque la probabilité des réponses considérées comme correctes aux signaux d'entrée dans lesquels cette situation peut être traduite sera devenue dominante[35] pour un nombre suffisant de signaux d'entrée. Turing suggère qu'une machine « apprenant » de cette manière à effectuer n'importe quelle tâche qui peut être réalisée par une machine universelle[36] sera en mesure de « faire de plus en plus de "choix" ou de "décisions" »[37], et que son comportement reposera ainsi sur des principes de plus en plus généraux, jusqu'à ce que ces principes soient suffisamment généraux pour que l'intervention directe de programmeurs ne soit plus nécessaire. Sans doute Turing a-t-il en vue, ici, une échelle de complexité des situations : parmi les nombreuses tâches complexes que la machine aura appris à réaliser, beaucoup seront plus ou moins proches les unes des autres ; face à une situation complexe nouvelle, la machine pourra essayer diverses réponses qu'elle possède déjà, en se fondant sur les signaux d'entrée communs à la nouvelle tâche et à celles qu'elle sait accomplir. La machine parviendra ainsi, à terme, comme un être humain, à trouver, pour la nouvelle tâche, une réponse qui pourra être tenue pour satisfaisante par des êtres humains. Certaines situations complexes peuvent être considérées comme composées elles-mêmes de situations complexes, de sorte que lorsque la machine saura « résoudre » un nombre suffisant de ces situations, elle sera également en mesure d'apporter une réponse – bonne ou mauvaise, là n'est pas la question - à la situation complexe de degré supérieur. La machine sera « éduquée » - c'est–à-dire qu'elle n'aura plus besoin d'intervention extérieure - lorsqu'elle aura atteint un degré suffisant de complexité. C'est alors, sans doute, qu'elle sera « sur-critique ».

La comparaison du comportement de la machine avec le processus humain de résolution d'un problème prend dans ce cadre une vigueur particulière : il ne s'agit plus simplement, pour la machine, de mettre en œuvre une configuration pour aboutir à un certain résultat, mais, partant d'un certain donné – les signaux d'entrée – de « trouver » la configuration déterminant une réponse qui sera considérée comme adéquate par des humains. Tout se passe, en somme, comme si la machine élaborait elle-même une démarche intellectuelle prenant en compte la donnée d'entrée.

Bref, Turing suggère la possibilité de constituer, par un processus « d'apprentissage », et en partant du niveau le plus élémentaire d'une machine inorganisée, la série ascendante de machines se simulant les unes les autres jusqu'à celle

35 Turing donne un exemple de machine inorganisée dans « Intelligent Machinery ». Il s'agit d'une machine constituée d'un nombre n d'unités semblables. Chaque unité dispose de deux entrées et une sortie. Celle-ci peut être ou non connectée à une entrée d'une ou plusieurs autres unités. Pour chaque entier r ($1 \leq r \leq n$), deux nombres, i(r) et j(r), sont choisis au hasard dans l'ensemble des entiers compris entre 1 et n. L'unité r est connectée aux unités i(r) et j(r). Les unités sont coordonnées par un dispositif qui émet des impulsions à des intervalles égaux, lesquels définissent des « moments ». A chaque moment, chaque unité peut avoir deux états. Chaque état est déterminé par le produit des états respectifs des unités à laquelle l'unité courante est connectée en entrée. Puisque les états d'une telle machine sont en nombre fini, son mouvement sera périodique, et sa période ne pourra pas être supérieure à 2^n. Ce type de machine peut enfin être compliqué en imaginant que les liens entre les unités soient eux mêmes constitués de telles unités. Or, une machine inorganisée peut être soumise à des « interférences », c'est-à-dire à certaines conditions imposées de l'extérieur, qui entraînent une modification de son comportement. Turing indique, en particulier, qu'il serait possible d'obtenir ces modifications en simulant le système « punitions-récompenses ». Si le comportement périodique de la machine n'est pas celui que l'on souhaite, un signal « punition » lui est envoyé, qui détermine un changement aléatoire de sa configuration. Les réponses « fausses » de la machine tendront à devenir statistiquement de plus en plus rares. Par là, une machine inorganisée peut être transformée en machine organisée, conçue dans un but déterminé. Plus précisément, Turing estime qu'une machine inorganisée peut être transformée en machine universelle, susceptible de simuler le comportement d'autres machines.

36 Peu importe que faire exécuter telle tâche par une machine soit ou non la meilleure façon de la réaliser.

37 « Intelligent Machinery », p. 126.

qui sera en mesure de faire bonne figure au jeu.

A travers l'hypothèse des « machines qui apprennent », le caractère ouvert du jeu de l'imitation devient décisif. Puisque le registre du jeu est l'ensemble des compétences humaines, on peut y parler de n'importe quoi, et l'examinateur doit, même, *a priori*, y parler de n'importe quoi. En fonction de cela, tout se passe comme si Turing admettait que la machine, pour passer le test de manière satisfaisante, devait être capable de fournir aux questions posées par ses adversaires lors du jeu une réponse qui sera considérée par eux comme « sensée » *y compris dans les cas non prévus par elle*. Le joueur mécanique du jeu de l'imitation doit, autrement dit, pouvoir répondre, dans ces derniers cas, sans que la machine « partielle » adéquate soit construite pour l'occasion, de l'extérieur, par une équipe de programmeurs ; elle doit pouvoir élaborer *elle-même* cette machine partielle, correspondant à tel cas non prévu[38]. D'où l'idée que l'apprentissage, tel que Turing le conçoit, est certainement la bonne méthode – voire la seule – pour construire une machine susceptible de l'emporter au jeu de l'imitation : une machine « éduquée » doit être en mesure de discuter avec un interlocuteur humain quelconque, comme le ferait lui-même un individu humain quelconque, c'est-à-dire de proposer, à partir d'une série de questions, des réponses diverses, qui ne seront peut-être pas satisfaisantes, mais dont il sera difficile de déterminer, dans un contexte tel que celui du jeu, si elles le sont radicalement moins que celles de ses interlocuteurs.

Dès lors, il sera permis de dire, non seulement que l'examinateur C du jeu se comporte *comme si*, pour lui, la machine A « comprenait », mais encore que A « comprend » effectivement, dans le même sens où cela peut être dit pour C. Pourtant, la méthode, ici, n'est pas celle de l'IA forte, du moins tant que celle-ci reste associée à la méthode classique consistant à programmer directement la machine.

La méthode classique consiste, en effet, à exposer formellement le programme qui régit entièrement le comportement de la machine, c'est-à-dire à écrire le texte, le « code », de ce programme. Par là, cette méthode présuppose un double niveau d'existence pour la machine : sous la forme d'un programme, c'est-à-dire d'une série de propositions bien formées en un certain langage, et sous la forme d'une suite d'assemblages matériels de pièces (par exemple des circuits imprimés), dont chaque élément – chaque « moment » - correspond à un état décrit par le programme. On postule alors un isomorphisme entre ces deux niveaux d'existence, de sorte qu'il sera toujours possible de fournir la description d'une machine considérée à un moment de son existence sous la forme d'un texte : le « code » correspondant à son action à ce même moment.

Or, qu'en est-il, dans ce contexte, de la « machine qui apprend » ? Celle-ci, puisqu'elle est « éduquée », a un « passé », une *histoire* ; cette histoire peut-elle être représentée par le code de la machine ? Imaginons une machine qui l'emporte au jeu de l'imitation après avoir été « éduquée ». Selon la méthode classique, cette machine devrait pouvoir être présentée sous la forme d'un ensemble d'instructions rédigées en un certain langage ; nous dirons, ici, pour la commodité du raisonnement, sous la forme d'un fichier texte où serait écrit son code. Ne peut-on imaginer, alors, qu'une seconde machine, absolument équivalente à la première, soit obtenue par simple copie du programme de celle-ci ? Nous pouvons supposer une machine universelle indéterminée à laquelle serait fourni le code élaboré par apprentissage pour une première machine ; cette seconde machine ne se comportera-t-elle pas exactement comme la première, puisque son code sera le même ? Ne bénéficiera-t-elle pas, en quelque sorte, du « passé » de la première sans avoir eu elle-même à « vivre » ce passé ?

En réalité, la question est celle de savoir ce que représente exactement le code que la seconde machine aura copié. Si le fichier ne comporte que le code de la machine « éduquée » correspondant à un certain moment de l'histoire de celle-ci – par exemple le moment où elle l'emporte au jeu de l'imitation – la seconde machine ne pourra être considérée comme un équivalent de la première, puisque l'essentiel du « passé » de celle-ci ne figurera explicitement nulle part dans le code copié. Le fichier texte comportera uniquement le code de certaines configurations fixées par la machine, et mises en jeu par les signaux d'entrée correspondant à la situation complexe, ou extrêmement complexe, en quoi consiste le moment où elle l'emporte au jeu. Il ne

38 On notera, ici, qu'il importe peu que la machine « partielle » réponde correctement ou non à la question posée ; elle doit simplement donner une réponse que l'examinateur, même s'il n'est pas d'accord, comprendra.

comportera pas nécessairement, pour représenter la machine à ce moment, le code d'autres configurations fixées par elle et correspondant à d'autres situations complexes rencontrées au cours de son « histoire ». Il conviendrait au moins, pour qu'il y ait duplication de la première machine par copie du code, que celui-ci comporte la description de chaque instant de la « vie » de la machine, c'est-à-dire, non seulement le code de toutes les configurations fixées par la machine, mais également celui de toutes les configurations modifiées par elle. Admettons même que cela soit possible, qu'un ingénieur prodigieusement laborieux ait rédigé le texte comportant la description de chaque instant de la « vie » d'une machine *x*, la copie du texte, son implémentation sur une machine universelle *y* (présumée vierge), permettra-t-elle de donner naissance à une seconde machine strictement équivalente à la première, ayant la même histoire, le même passé, et devant avoir, face à la même situation complexe, le même comportement qu'elle ? Nous avons vu que, selon Turing, l'apprentissage, pour la machine, consistait en un processus au cours duquel la probabilité des actions non souhaitées diminue ; or, cela ne nous dit rien de la probabilité des actions qui ne sont pas directement traitées par l'apprentissage. Cette probabilité reste prise dans la logique de la « surcriticalité », c'est-à-dire de l'explosion combinatoire. Dès lors, nous ne pouvons pas être assurés, dans le cas où une machine *y* disposerait du même programme « total » qu'une machine « éduquée » *x*, que ces deux machines répondront de manière identique à la même question. Le fichier texte fournirait, au mieux, le texte du code correspondant à chaque instant de la vie de *x*, c'est-à-dire uniquement le texte de ce qui a été réalisé par celle-ci, et non celui de toutes les actions possibles dans le cadre d'une explosion combinatoire. Rien ne garantit que deux machines pouvant être décrites, jusqu'à une certaine étape, à l'aide du même code, réagiront exactement de la même façon à l'étape suivante ; elles pourront adopter l'une et l'autre des comportements à la probabilité proche et cependant différents.

En vérité, une machine *y* constituée à l'aide du code d'une machine *x*, et qui, par là même, lui aurait en quelque sorte « volé son histoire », vivrait, sitôt qu'elle aurait commencé d'exister, sa *propre* histoire. Bien plus, sa naissance spécifique, par copie, constituerait déjà, s'agissant de cette histoire, une singularité, un événement qui ne pourrait figurer dans le code copié. En tout état de cause, le fichier texte complet, comportant la description de chaque instant de la « vie » d'une machine serait sans doute une description de « l'histoire » de celle-ci, mais le récit d'une histoire constitue lui-même une autre histoire que celle qu'il décrit. Bref, le statut d'existence du fichier texte et celui de la machine en tant qu'entité ayant un « passé » ne peuvent être les mêmes.

On l'a vu, la structure particulière du jeu de l'imitation interdit qu'on remplace la question « les machines peuvent-elles penser ? » par la question « les machines peuvent-elles comprendre ? ». Le jeu implique trois protagonistes et n'est pas réductible à un échange de parole indifférencié entre deux interlocuteurs ; l'échange de paroles qu'il met en scène ne peut être considéré indépendamment du but qu'il s'agit pour les joueurs de viser ou d'interdire : distinguer un homme d'une machine, ce qui implique de reconnaître de la pensée *pour* reconnaître un homme.

En outre, à travers la problématique de l'apprentissage, c'est tout à la fois la capacité de la machine à se prendre pour objet et la relation de cette faculté à son devenir qui sont mis en scène.

La méthode attachée à la théorie de l'IA forte telle que l'énonçait Searle ne permet pas de rendre compte, de cette structure singulière du jeu, ni de cet appel, dans la réflexion de Turing, au « passé » de la machine victorieuse au jeu de l'imitation. Celui-ci n'éclaire pas ce qu'est le penser pour l'homme ; il ne débouche pas, par exemple, sur la possibilité d'une analyse du terme « penser » qui conduirait à contenir strictement le sens de celui-ci dans les limites de la notion d' « état cognitif ». Il montre, bien plutôt, que le comportement d'une machine universelle peut être tel qu'il renvoie à la même opacité conceptuelle, exprimée par le terme « penser », que le comportement intellectuel de l'être humain. Le problème de la pensée, qui ne se pense que comme principe d'intelligibilité, n'est ni résolu, *ni rejeté* par les hypothèses de Turing, comme si le comportement intellectuel de l'être humain pouvait être ramené strictement au fonctionnement de la machine ; c'est bien plutôt ce fonctionnement même qui, au travers des hypothèses de Turing, se révèle porteur possible de l'interrogation constitutive du terme « penser » lui-même. Le jeu de l'imitation se construit sur la dimension *spéculative* de celui-ci, et conduit, non pas au refus de cette dimension, mais à la nécessité de « faire

avec » en prenant en compte le fait qu’elle enveloppe la situation d’échange de paroles, quand bien même cet échange impliquerait une machine.