

# Wie Maschinen denken

---

THOMAS HAINSCHO

In der Dokumentation *Technocalyps* (2006) stellt Joseph M. Rosen, Professor of Surgery sowie Adjunct Professor of Engineering, das Problem einer Rede von denkenden Maschinen deutlich dar:

At first we say that if a computer could play chess, then it will think like us. And then we get a computer to play chess and we say that is really not thinking. And the answer is that we don't really know what thinking is. I would argue that machines do a pretty good job right now at thinking. And they don't do as good a job at creating, although we don't really know what creating is. And they don't do as good a job at having a soul, but we don't really know what a soul is. But when we can define it, they do a pretty good job at doing it. (TECHNOCALYPS: 2006, Interview with Joseph M. Rosen, 21m24s)

Um zu entscheiden, ob eine bestimmte Maschine so etwas wie Gedanken hat oder nicht, sowie bei dem Urteil darüber, ob Maschinen überhaupt denken können, muss man sich der Frage dieses Sammelbands stellen: Was ist das Denken? Darüber hinaus stellt sich noch die Frage, was eine Maschine ist, etwa in Abgrenzung zum Menschen, zum Tier oder zur unbelebten Natur. Werden Definitionen des Denkens gegeben, die Schritt für Schritt beschreiben, was beim Denken passiert, dann können Maschinen gebaut – das heißt Computerprogramme geschrieben – werden, die Denkvorgänge gemäß diesen Definitionen simulieren. Allerdings tritt angesichts der Simulation von Denkvorgängen häufig die von Rosen angesprochene Enttäuschung auf. Denksimulationen scheinen nur einen eingeschränkten Bereich

des Denkens zu erfassen und das Phänomen des Denkens entzieht sich seiner Festlegung: We say that is not really thinking.

Ich möchte im Folgenden beschreiben, zu welchen Operationen Maschinen fähig sind und inwiefern diese Operationen als Denken verstanden werden können. Im ersten Teil möchte ich den Begriff der Maschine klären und einen Begriff des Denkens nach ihrem Vorbild herausarbeiten. Im zweiten Teil möchte ich Konsequenzen nennen, die entstehen, wenn ein Begriff des Denkens, der nach dem Vorbild der Maschine gebildet ist, für die Erklärung des menschlichen Denkens herangezogen wird.

## **DENKENDE MASCHINEN**

### **Unterscheidung von Mensch und Maschine – Imitation-Games und Turing-Tests**

Einer der bekanntesten Versuche, zu entscheiden, ob Maschinen denken können, stammt von Alan M. Turing. Einleitend in seinem Aufsatz *Computing Machinery and Intelligence* (1950) lehnt Turing allerdings die Behandlung der Frage, ob Maschinen denken können, ab. Eine Antwort auf diese Frage bedürfte einer umfassenden Begriffsklärung und anstatt sich an Definitionen der Begriffe ‚Maschine‘ und ‚Denken‘ zu versuchen, schlägt Turing vor, die Frage durch eine andere zu ersetzen. Er entwirft eine Versuchsanordnung – das Imitation-Game –, die entscheiden soll, ob Maschinen denken können oder nicht.

Im Imitation-Game unterhalten sich eine Fragestellerin bzw. ein Fragesteller C schriftlich ohne Sicht- oder Hörrkontakt mit einem Mann A und einer Frau B. Die Aufgabe von C ist es, herauszufinden, wer von den beiden anderen teilnehmenden Personen P<sub>1</sub> und P<sub>2</sub> der Mann A, und wer die Frau B ist. Das einzige Hilfsmittel dafür sind schriftliche Antworten von A und B auf schriftliche Fragen von C. Diese Fragen kann C beliebig an P<sub>1</sub> und P<sub>2</sub> stellen. Das Ziel von A ist es, C davon zu überzeugen, dass er die Frau sei. Das Ziel von B ist, C davon zu überzeugen, dass sie die Frau sei. Die Kommunikation verläuft ausschließlich schriftlich, das ideale Setting für das Imitation-Game wäre ein Teleprinter, sodass für C keine Rückschlüsse über die Handschrift möglich sind. (Vgl. Turing 1950: 434)

Nach Schilderung des Ablaufs und der Ziele schlägt Turing eine modifizierte Version des Spiels vor, bei dem eine Maschine an die Stelle des Mannes A tritt. Diese Veränderung des Spiels stellt die Grundlage für den sogenannten Turing-Test dar, jenem nach Turing benannten Experiment, anhand dessen entschieden werden soll, ob ein Computer denken kann oder nicht. Turing (1950: 434) stellt die Frage: „Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman?“ Mit dieser Frage soll die ursprüngliche Frage, ob Maschinen denken können, ersetzt werden. Aus Turings Aufsatz geht nicht hervor, ob es sich bei dem modifizierten Imitation-Game um ein Frage-Antwort-Spiel handelt zwischen (1) einem Menschen und einer Maschine, die vorgibt, ein Mensch zu sein, oder zwischen (2) einer Frau und einer Maschine, die vorgibt, eine Frau zu sein. Ein Großteil der Rezeption vertritt die erste Interpretation (vgl. Oppy & Dowe 2011).

Dem modifizierten Imitation-Game geht keine Begriffsanalyse des Denkens voraus und es handelt sich um kein Verfahren, das Denkfähigkeit von bestimmten mentalen Phänomenen abhängig macht. Vielmehr schlägt Turing vor, dass es keinen Anlass dafür gibt, sich länger den Kopf über die Denkfähigkeit von Maschinen zu zerbrechen, wenn einer Maschine die Täuschung – die Imitation – gelingt und sie aus Sicht von C das erfüllt, wofür Denken eine notwendige Voraussetzung ist.

Im Vergleich zu kognitionswissenschaftlichen Experiment-Designs ist das Imitation-Game erstaunlich ungenau. Stützt man sich auf Turings „own beliefs in that matter“ (Turing 1950: 442), lassen sich allerdings implizite Testbedingungen formulieren: Er glaubt, dass am Ende des 20. Jahrhunderts die Wahrscheinlichkeit  $p$  für das richtige Urteil einer bzw. eines „average interrogator“ nach fünf Minuten Befragung  $p \leq 0,7$  sei, das heißt, dass dem Computer in mehr als 30% der gespielten Imitation-Games die Täuschung gelingen muss, um zu gewinnen. Für einen tatsächlich durchgeführten Test müssen trotzdem noch zahlreiche Ergänzungen zu Turing vorgenommen werden: Beispielsweise lässt er offen, ob der Fragestellerin bzw. dem Fragesteller eingegebener Text erhalten bleibt, ob die menschliche, Fragen-beantwortende Person bestimmte Gebote oder Verbote hat (zum Beispiel, dass sie nicht schweigen oder selbst Fragen stellen darf), und schließlich auch, wie die Fragestellerin bzw. der Fragesteller urteilt:

Wie werden Glückstreffer verhindert; muss die Entscheidung begründet werden; ist es auch möglich, mit *Ich weiß es nicht* zu antworten etc.

Eine Kritik des Turing-Tests ist nicht das Thema dieses Texts, ich möchte aber darauf hinweisen, dass ungeachtet dieser Probleme seit der Veröffentlichung von *Computing Machinery and Intelligence* vor rund 65 Jahren zahlreiche Turing-Tests in verschiedenen Varianten durchgeführt worden sind, und immer wieder verkündet wurde, dass ein Computerprogramm den Turing-Test bestanden hätte und Maschinen ab sofort denken können. Zuletzt erregte ein Turing-Test, der 2014 an der Royal Academy in London durchgeführt wurde, mediale Aufmerksamkeit (vgl. Warwick & Shah 2015). 30 Fragesteller\_innen führten fünf jeweils fünf Minuten lange Gespräche mit fünf verschiedenen Chat-Bots sowie einem jeweils anderen Menschen. Bei einem der fünf Chat-Bots konnten zehn der 30 Fragesteller\_innen nicht zwischen menschlichen Antworten und den Antworten des Computerprogramms unterscheiden, das heißt 33% der Zuordnungen waren falsch. Warwick und Shah (2015: 15f.) haben sich an Turings 30%-Grenze orientiert und meinen, dieser Test sei zugunsten der Maschine ausgegangen. Sie weisen allerdings auch auf Schwierigkeiten hin, richtige und falsche Identifikationen in einem Turing-Test auszuwerten: In dem beschriebenen Test wurden von den zehn als Fehlinterpretationen gedeuteten Frage-Antwort-Spielen die Antworten des Chat-Bots in sechs Fällen für menschliche Antworten gehalten und in vier weiteren Fällen waren sich die Fragesteller\_innen unklar darüber, ob die Antworten von einem Menschen oder einer Maschine stammen. Jene Fälle, in denen die Antworten der menschlichen Versuchsteilnehmer\_innen für die Antworten der Maschine gehalten wurden, sind in den zehn als Fehlinterpretationen gedeuteten Frage-Antwort-Spielen nicht mitgezählt.

## **Maschinendenken – Universalrechner und Turingmaschinen**

Es gilt heute als selbstverständlich, dass Computer die einzige Art von Maschine sind, die das Imitation-Game spielen können. Mitte des 20. Jahrhunderts war dieser Umstand noch klärungsbedürftig und Turing (1950: 436) schreibt, dass Digitalrechner – „digital computers“, einleitend auch als „electronic computer“ bezeichnet – dazu in der Lage sind, am Spiel teilzunehmen. Seine vorangegangenen Schriften lässt er dabei unerwähnt, ob-

wohl das Modell, das er für Digitalrechner vor Augen hatte, zweifelsohne die sogenannte Turingmaschine war. Ich möchte im Folgenden zwei mögliche Gründe für Turings Einführung von Digitalrechnern geben.

Erstens nehme ich an, dass es Turing wichtig war, eine konstruierbare Maschine in das Imitation-Game einzuführen. Die Turingmaschine ist ein theoretisches Modell, mit ihr wäre das Imitation-Game nur als Gedankenexperiment möglich, und ich denke, dass sich Turing das Imitation-Game als tatsächlich durchführbares Experiment vorgestellt hat. Er nennt Konstruktionsbedingungen für Imitation-Game-Maschinen (vgl. Turing 1950: 435f.) und erwähnt später im Text auch die „Manchester machine“ (Turing 1950: 441), einen digitalen Großrechner der University of Manchester, an dessen Entwicklung er seit 1948 beteiligt war (vgl. Hodges 2014: 471ff.).

Der zweite Grund liegt in einer möglichen Analogie zwischen dem beschriebenen Aufbau des Digitalrechners und dem menschlichen Gehirn. Es ist nicht abwegig anzunehmen, dass Turing in dieser Rechnerarchitektur eine Analogie zu seiner Vorstellung des menschlichen Gehirns sah. Die von Turing (1950: 437) genannten Komponenten des Digitalrechners – Speicher, ausführende Einheit und Kontrollwerk („store“, „executive unit“, „control“) – entsprechen den Komponenten der heute als Von-Neumann-Architektur bezeichneten Rechnerarchitektur. Diese Bauweise geht auf John von Neumann zurück und wurde schriftlich erstmals 1945 in seinem Text *First Draft of a Report on the EDVAC*, einem Teil der Projektdokumentation zum Bau des Großrechners EDVAC, beschrieben (Neumann 1992). Darin beschreibt von Neumann die drei Komponenten, aus denen sich ein „high speed automatic digital computing system“ (Neumann 1992: 1, §1.1) zusammensetzt: der zentralen Recheneinheit („central arithmetical part“ – CA, deren Aufgaben entsprechen den Aufgaben der ausführenden Einheit bei Turing), der zentralen Kontrolleinheit („central control“ – CC, deren Aufgaben entsprechen den Aufgaben von Turings Kontrollwerk) sowie dem Speicher („memory“ – M, entsprechend Turings Speicher) (vgl. Neumann 1992: 1–3, §§2.1–5). In dieser Architektur gibt es einen gemeinsamen Speicher für Programme, Befehle und Daten. Diese Bauweise wird als das Konzept des Universalrechners verstanden, nach dem auch moderne Computer gebaut werden. Das Universale an dieser Art von Maschine ist, dass sie Aufgaben lösen kann, die nicht alleine von ihrer Bauweise abhängig sind, sondern vom Inhalt ihres Speichers. Der gemeinsame Speicher ermöglicht, der Rechneranlage nur durch das Ändern ihres Speicherinhalts

andere Aufgaben zu geben, ohne sie dafür umzubauen. Rechneranlagen der 20er- und 30er-Jahre wurden für wenige, mitunter auch nur für eine Aufgabe konstruiert. Der Großrechner Differential Analyzer wurde Ende der 1920er-Jahre am MIT gebaut, nur um Differentialgleichungen zu lösen (vgl. Petzold 2008: 163f.). Bis in die 1950er-Jahre war nicht klar, dass der Universalrechner das dem modernen Computer zugrundeliegende Modell werden sollte.

Entsprechend einer Maschine, die für keinen bestimmten Zweck gebaut ist, sondern ihre Aufgabe abhängig vom Inhalt ihres Speichers ausführt, kann auch der Mensch als Universalrechner verstanden werden: Die Handlungen eines Menschen mögen in der Art von den Inhalten seines Gehirns abhängig sein, wie die ausgeführten Befehle eines Computers von seinem Speicherinhalt abhängen. Computer können ihren Speicher umschreiben, Menschen ihre Gedanken ändern. Die Gehirn-Analogie ist als bildlicher Vergleich auch bei von Neumann zu lesen: „The three specific parts CA, CC (together C) and M correspond to the *associative* neurons in the human nervous system“ (Neumann 1992: 3, §2.6; vgl. auch §4.2, §6.3).

Hinweise dafür, dass Turing dieser Interpretation gegenüber wohlwollend eingestellt war, findet man sowohl in seiner Biographie als auch in seinen Texten. So schreibt Andrew Hodges in seiner Turing-Biographie *Alan Turing: The Enigma*, dass Turing gesagt hat, er beschäftige sich mit dem Bau von Gehirnen (vgl. Hodges 2014: 364f.). Neben der Zuschreibung von Intelligenz an Maschinen in vielen Texten Turings (1950, 1996 u.a.), gibt es einen Hinweis darauf, dass er von Neumanns vorgeschlagene Bauweise des Universalrechners kannte und seiner Gehirn-Analogie zustimmt. Dass Turing von Neumanns Text kannte, ist belegt durch einen Verweis in *Proposed Electronic Calculator*, einem vermutlich 1945 verfassten, unveröffentlichten Entwurf zur Automatic Computing Engine (ACE) (Turing o.J.: 3), einer Rechneranlage, die Turing ab 1944 in London mitkonstruiert hat (vgl. Hodges 2014: 435ff.; Petzold 2008: 167). In diesem Entwurf verwendet Turing auch dieselben Bezeichnungen für Ein- und Ausgabeschnittstellen wie von Neumann („input organ“ und „output organ“) (vgl. Turing o.J.: 12; Neumann 1992: 3, §2.6).

Ich komme nun zurück auf die Maschine, die am Imitation-Game teilnehmen kann. An die Beschreibung des Digitalrechners wird in *Computing Machinery and Intelligence* mit dem Beweis der Universalität des Digitalrechners angeschlossen. Turing (1950: 439–442) beweist die Universalität

des Digitalrechners, indem er die Äquivalenz zwischen Digitalrechner und endlichem Zustandsautomaten zeigt. Die Eigenschaft der Universalität, mit der auch die Analogie zwischen Gehirn und Digitalrechner begründet wurde, soll im Folgenden noch näher betrachtet werden. Ihre Erörterung soll auf die Gleichsetzung der Denkfähigkeit von Mensch und Maschine hinführen. Von Neumann ist zwar der Namensgeber für die Bauweise des Universalrechners und gilt als dessen Erfinder (vgl. Herold, Lurz & Wohlrab 2006: 34), sein auch in Nachschlagewerken häufig unerwähnter Vorgänger ist allerdings Turing selbst, der den Begriff der ‚universal machine‘ bereits 1936/37 in *On computable numbers, with an application to the Entscheidungsproblem* einführt. Es handelt sich bei Turings ‚universal machine‘ allerdings nicht um eine Maschine, die gebaut werden kann, sondern um eine ideelle Maschine, die ein Modell für Berechenbarkeit darstellt.

Turing entwickelte in *On computable numbers, with an application to the Entscheidungsproblem* das Konzept der Turingmaschine. Er selbst hat sein Konzept darin nur als ‚computing machine‘ bezeichnet, der Name Turing machine war erstmals 1937 in einer Besprechung von Turings Text im *Journal of Symbolic Logic* durch Alonzo Church (1937: 43) zu lesen (vgl. Petzold 2008: 63). Eine Turingmaschine besteht konzeptuell aus sechs Elementen, die wie folgt notiert werden (ich folge einer vereinfachten Schreibweise von Hopcroft, Motwani & Ullman 2002):

$M = (Q, \Sigma, \delta, q_0, \#, F)$ , wobei

- $Q$  die endliche Menge an Zuständen ist, die  $M$  einnehmen kann,
- $\Sigma$  die Menge an Zeichen ist, die  $M$  einlesen und schreiben kann,
- $\delta$  die Überführungsfunktion ist, die dafür verantwortlich ist, dass aus einem Zustand sowie einem eingelesenen Zeichen ein nächster Zustand, ein zu schreibendes Zeichen sowie entweder eine Bewegung nach links (L) bzw. rechts (R) oder keine Bewegung (0) folgt:  $Q \times \Sigma \rightarrow Q \times \Sigma \times \{L, R, 0\}$ ,
- $q_0 \in Q$  der Startzustand ist,
- $\# \in \Sigma$  das Leerzeichen ist, das in allen leeren Feldern am Band steht und
- $F \subset Q$  die Menge von Endzuständen bildet.

In der Regel wird eine Turingmaschine bildlich dargestellt als ein in Felder oder Zellen gegliedertes, unendlich langes Band mit einer Schreib/Lese-Vorrichtung, die den Inhalt der Zellen lesen und verändern kann. Man kann

etwa an eine Schreibmaschine mit Band denken, bei der ein Schreibmaschinenkopf Zeichen auf das Band schreiben, vom Band lesen und löschen kann. Eine Turingmaschine ist zu jedem Zeitpunkt in genau einem Zustand  $q$  der Zustandsmenge  $Q$ . Am Anfang befindet sie sich im Startzustand  $q_0$  und liest – nicht per Definition, aber per Konvention – den Inhalt des zweiten Feldes am Band  $f_2$  ein. In  $f_2$  steht ein Zeichen  $\sigma$  aus der Zeichenmenge  $\Sigma$  und mit dem aktuellen Zustand  $q_0$  sowie dem eingelesenen Zeichen  $\sigma$  lässt sich über  $\delta$  der nächste Schritt der Maschine ermitteln:  $\delta(q_0, \sigma)$  ergibt einen Nachfolgezustand  $q_n \in Q$ , ein neues Zeichen  $\sigma \in \Sigma$  für das aktuell eingelesene Feld  $f_2$  sowie entweder eine Bewegung zum links angrenzenden Feld  $f_1$ , zum rechts angrenzenden Feld  $f_3$  oder keine Bewegung (die Schreib/Lese-Vorrichtung bleibt auf Feld  $f_2$  stehen). Die Turingmaschine hört auf, sich zu bewegen, wenn ein erreichter Zustand  $q_n$  der Menge der Endzustände  $F$  angehört. In diesem Fall spricht man davon, dass die Maschine hält. Gelangt eine Maschine nie in einen Endzustand – etwa, weil sie immer wieder dieselben Schritte durchführt –, spricht man davon, dass die Maschine nicht hält.

Eine typische Aufgabe für eine Turingmaschine ist etwa, eine Zahl, die am Band steht, einzulesen und anschließend ein Zeichen zu schreiben, das von einer Eigenschaft dieser Zahl abhängig ist. Komplexe Turingmaschinen entscheiden etwa, ob es sich bei dem Bandinhalt um eine Primzahl handelt, eine Fibonacci-Zahl oder um ein Element, das einer bestimmten Menge angehört etc. Ein einfaches Beispiel ist eine Turingmaschine, die entscheidet, ob es sich beim Bandinhalt um eine gerade oder ungerade Zahl (hier in unärer Codierung, also als Repräsentation des Zahlenwerts mithilfe eines Zeichens „1“) handelt: für den Bandinhalt #I###... hält die Turingmaschine mit #Iu###..., bei #II###... mit #IIg###..., bei #III###... mit #IIIu###... etc. Um das Ergebnis der Turingmaschine zu verstehen, muss die Zeichencodierung geklärt sein: dass es sich um unär codierte Zahlen handelt, dass „g“ als gerade interpretiert wird und „u“ als ungerade. Es wäre auch möglich, dass die Turingmaschine Zahlen in dezimaler Codierung, also als 1, 2, 3, ... verarbeitet, dafür wären aber wesentlich mehr Zustände und eine komplexere Überführungsfunktion  $\delta$  nötig als bei einer unären Codierung.

Die Universalität der Turingmaschine besteht darin, dass es eine Turingmaschine gibt, die jede andere Turingmaschinen simulieren kann: Darauf hinleitend werden im fünften Abschnitt von Turing (1936/37: 240)

die Konzepte der Standardbeschreibung einer Turingmaschine, „standard description (S.D)“, und der Beschreibungszahl, „description number (D.N)“, eingeführt. Standardbeschreibungen sind eine Codierung der Überführungsfunktion  $\delta$  sowie der Zeichen- und Zustandsmenge in eine Folge aus sieben verschiedenen Zeichen. Eine Beschreibungszahl ist die eindeutige Repräsentation einer Turingmaschine als Folge der Zahlen von 1 bis 7, das heißt als Zahl: „[...] we shall have a description of the machine in the form of an arabic numeral“ (Turing 1936/37: 240). Es ist möglich, die Standardbeschreibung einer Turingmaschine  $M_n$  als Inhalt auf das Band einer anderen Turingmaschine  $M$  zu schreiben. Wenn  $M$  startet, liefert sie dasselbe Ergebnis wie  $M_n$ , bloß braucht sie dafür nur jene Zustände, die es erlauben, Standardbeschreibungen zu dechiffrieren: „It is possible to invent a single machine which can be used to compute any computable sequence. If this machine  $U$  is supplied with a tape on the beginning of which is written the S.D of some computing machine  $M$ , then  $U$  will compute the same sequence as  $M$ .“ (Turing 1936/37: 241–242)

Damit sollte klar geworden sein, warum Turing der theoretische Vorgänger für von Neumann war. Die Architektur des Universalrechners (der gebaut werden kann) gleicht konzeptuell der universellen Turingmaschine (die nicht gebaut werden kann). Mithilfe der universellen Maschine zeigt Turing, dass das Entscheidungsproblem nicht von einer Turingmaschine gelöst werden kann. Mit Entscheidungsproblem wird die Frage bezeichnet, ob es für eine beliebige Aussage in bestimmten formalen, widerspruchsfreien Kalkülsystemen ein Berechnungsverfahren gibt, das diese Aussage als wahr beweist oder als falsch widerlegt. Aussagen, die bewiesen bzw. widerlegt werden können, werden entscheidbar genannt. Kurt Gödel hatte bereits 1930 mit dem Unvollständigkeitssatz gezeigt, dass es Aussagen gibt, die nicht aus den Grundsätzen des Kalkülsystems in der *Principia Mathematica* abgeleitet werden können. Solche Aussagen können innerhalb des Formalismus der *Principia Mathematica* weder als wahr bewiesen noch als falsch widerlegt werden, sind also unentscheidbar. Turings Beweis zeigt, dass es kein allgemeines Berechnungsverfahren gibt, das über die Entscheidbarkeit einer Aussage entscheidet. Er zeigt, dass es keine universelle Turingmaschine gibt, die entscheidet, ob eine beliebige simulierte Maschine jemals das Zeichen „0“ auf das Band schreibt.

Wie Petzold (2008: 325) schreibt, wurde erst rund 20 Jahre nach der Veröffentlichung von *On computable numbers, with an application to the*

*Entscheidungsproblem* klar, dass die Turingmaschine ein Modell für Berechenbarkeit darstellt. Mit ihr lässt sich eine Theorie der Berechenbarkeit bilden, in der es um die Identifikation von entscheidbaren und unentscheidbaren Problemen sowie deren Eigenschaften und Modellierung geht. Turing konnte zeigen, dass es Probleme gibt, die nicht von Maschinen gelöst werden können – zumindest nicht von Maschinen, die nach dem Modell der Turingmaschine funktionieren, und bis heute sind keine anderen Maschinen bekannt. Die Probleme, die von Turingmaschinen gelöst werden können, bilden eine Menge, deren Elemente alle die Eigenschaft haben, entscheidbar zu sein. Es ist bewiesen, dass die Menge der für Turingmaschinen entscheidbare Probleme äquivalent ist zu der Menge von Problemen, die mit Hilfe anderer Berechenbarkeitsmodelle entschieden werden kann. Das heißt Turingmaschinen sind hinsichtlich ihrer Berechenbarkeit gleichmächtig wie etwa Gödelsche rekursive Funktionen, das  $\lambda$ -Kalkül von Alonzo Church und Stephen Kleene oder endliche Zustandsautomaten (vgl. Petzold 2008: 327). Eine Lösung, die von einer Turingmaschine für ein Problem gefunden wird, lässt sich auch mithilfe eines der anderen genannten Modelle finden. Ein Problem, das mit einer Turingmaschine nicht gelöst werden kann, lässt sich auch mit keinem dieser anderen Konzepte lösen. Jeder heute bekannte Computer lässt sich auf einer Turingmaschine simulieren, das heißt, für jedes Programm, das auf einem heute bekannten Computer ausgeführt wird, kann eine äquivalente Turingmaschine konstruiert werden. Somit sind Computer – seien sie auch noch so leistungsstark – nicht in der Lage, unentscheidbare Probleme zu lösen. Der Berechenbarkeit eines Computers sind aufgrund seines endlichen Speichers sogar engere Grenzen gesetzt. Eine Turingmaschine kann mit allen Zahlen operieren, während Computer nur mit jenen Zahlen operieren können, die klein genug sind, um in den Speicher zu passen.

Um zur Gleichsetzung der Denkfähigkeit von Mensch und Maschine zu kommen, gibt es noch einen weiteren Punkt, auf den ich eingehen möchte, Turings Beitrag zur Church-Turing-These im neunten Abschnitt von *On computable numbers, with an application to the Entscheidungsproblem*. Der Name Church-Turing-These soll nach Stephen Kleene (1967: 232) für die Churchsche These verwendet werden, wenn Berechenbarkeit im Zusammenhang mit Turingmaschinen besprochen wird (vgl. Petzold 2008: 327). Alonzo Church und Turing haben voneinander unabhängig an einer Formalisierung des Begriffs der Berechenbarkeit gearbeitet und über eine

Lösung des Entscheidungsproblems nachgedacht, die sie beide jeweils 1936 veröffentlicht haben: Church im März 1936 in *A note on the Entscheidungsproblem* und Turing in *On computable numbers, with an application to the Entscheidungsproblem*, das in zwei Teilen Oktober 1936 und April 1937 erschien (Petzold 2008: 62f.). Die sogenannte church'sche These wurde 1952 durch Kleene formuliert und lautet: „Every effectively calculable function (effectively decidable predicate) is general recursive.“ (Kleene 1971: 300) Diese These stützt sich auf den Beweis der Äquivalenz zwischen  $\lambda$ -Kalkül und Gödelschen generell rekursiven Funktionen und behauptet, dass sich jede, einem intuitiven Verständnis nach berechenbare Funktion generell rekursiv, d.h. formal berechnen lässt. Diese These gilt auch für Turingmaschinen, da es auch bewiesen ist, dass die Menge von Problemen, die eine Turingmaschine lösen kann, äquivalent ist zu der Menge von Problemen, die von anderen Berechenbarkeitsmodellen gelöst werden kann. Die Annahme, dass zu diesen anderen Berechenbarkeitsmodellen auch menschliche Rechenfähigkeiten gehören, ist eine These, die nicht bewiesen werden kann, weil menschliche Rechenfähigkeit nicht hinreichend genau definiert werden kann. Diese Annahme wird von Turing mithilfe dreier Argumente verteidigt, die er allerdings wie folgt einschränkt: „All arguments which can be given are bound to be, fundamentally, appeals to intuition, and for this reason rather unsatisfactory mathematically.“ (Turing 1936/37: 249)

Im ersten Argument, „a direct appeal to intuition“ (Turing 1936/37: 249), beschreibt Turing einen rechnenden Menschen, der mit Stift und Papier arbeitet. Er weist darauf hin, dass es nur endlich viele verschiedene Symbole sein können, mit denen der Mensch arbeitet, dass der Mensch zu jedem Zeitpunkt in einem bestimmten Geisteszustand (im Original: „state of mind“, unter Anführungszeichen) sei, und dass sein Verhalten von seinem Geisteszustand sowie am Papier zu lesenden Symbolen abhängig ist (vgl. Turing 1936/37: 249–252). Die Rechenschritte und Handlungen des Menschen werden so lange abstrahiert und mit Analogien beschrieben, bis sie den Operationen einer Turingmaschine gleichen. Es bleibt nichts mehr übrig, worin sich die Rechenfähigkeit des Menschen von denen der Turingmaschine unterscheiden kann. Somit sei auch der Mensch hinsichtlich seiner Rechenfähigkeit äquivalent zur Turingmaschine. Petzold (2008: 189) zitiert Hodges (2014: 133), für den dieses erste Argument „the most unusual ever offered in a mathematical paper“ sei. Dieser Einschätzung ist

Recht zu geben: Für die Zielsetzung von Turings Text – zu zeigen, dass das Entscheidungsproblem nicht gelöst werden kann – scheint die Äquivalenz zwischen menschlichen und maschinellen Rechenfähigkeiten irrelevant zu sein. Es mag den Umfang der berechenbaren Zahlen – *The extent of the computable numbers* ist der Titel des neunten Abschnitts – klären, doch wozu? Eine Antwort darauf möchte ich im nächsten Abschnitt geben.

## **Gleichsetzung von Mensch und Maschine – Science-Fiction und äquivalente Berechenbarkeit**

Kommen in der Literatur oder im Film Maschinen vor, denen die Fähigkeit zu denken zugesprochen wird, handelt es sich häufig um besondere Maschinen, die kaum Ähnlichkeit mit bekannten Maschinen haben. Etwa wird im Film *Ex Machina* (2015) das Gehirn der vermeintlich denkfähigen Maschine gezeigt, das bläulich leuchtet. Es besteht nicht aus elektrischen Schaltungen, sondern aus Gel, dessen molekularer Aufbau sich selbst manipulieren kann, es ist Wetware und keine Hardware. Weder die Terminators in den gleichnamigen Filmen (1984–2015) noch der Computer HAL 9000 im Film *2001: A Space Odyssey* (1968) sehen so aus wie bekannte Maschinen oder Computer. Damit wird der Fiction-Aspekt von Science-Fiction stärker betont als der Science-Aspekt.

Dennoch ist etwa Arthur C. Clarke, der Autor von *2001: A Space Odyssey* (1968), bemüht, seinen Science-Fiction-Roman möglichst nah an der wissenschaftlichen Forschung anzulegen. Er gibt darin einen kurzen Exkurs über die Entwicklung der künstlichen Intelligenz:

Then, in the 1960s, solid-state microelectronics had been perfected. With its advent, it was clear that artificial intelligences at least as powerful as Man's need be no larger than office desks. [...]

In the 1980s, Minsky and Good had shown how neural networks could be generated automatically – self replicated – in accordance with any arbitrary learning program. Artificial brains could be grown by a process strikingly analogous to the development of a human brain. In any given case, the precise details would never be known, and even if they were, they would be millions of times too complex for human understanding. Whatever way it worked, the final result was a machine intelligence that could reproduce – some philosophers still preferred to use the word ‚mimic‘ –

most of the activities of the human brain – and with far greater speed and reliability. (Clarke 2000: 117)

Clarke verweist in dieser Passage auf den KI-Forscher Marvin Minksy und den Kryptologen Irving John Good – letzterer hat mit Turing auch an der Entschlüsselung der Enigma gearbeitet (vgl. Hodges 2014: 249ff.). In der zitierten Entwicklungsgeschichte von den Anfängen der künstlichen Intelligenz bis zum Supercomputer HAL 9000 gibt es keine unbekannten Komponenten. Die künstlichen neuronalen Netzwerke stehen in einer Entwicklungslinie zur Mikroelektronik der 60er-Jahre sowie zum ebenfalls im Text erwähnten Großrechner ENIAC der 40er-Jahre (vgl. Clarke 2000: 117).

Damit ist bei Clarke implizit der Hinweis auf die Gleichsetzung der Denkfähigkeit von Mensch und Maschine enthalten: Eine Maschine, die denken kann, muss entweder hinsichtlich ihrer Berechenbarkeit äquivalent zur Turingmaschine sein oder sie ist eine fiktive Maschine. Wenn nicht-fiktive Maschinen denken können und es nicht-fiktive Maschinen gibt, die hinsichtlich ihrer Berechenbarkeit äquivalent zur Turingmaschine sind, dann können alle Turing-äquivalenten Modelle für Berechenbarkeit denken.

Unter dieser Voraussetzung hat die Church-Turing-These gravierenden Einfluss auf die Auffassung der Denkfähigkeit von Menschen: Wenn Menschen denken können und die Church-Turing-These stimmt, dann muss allen Turing-äquivalenten Konzepten dieselbe Denkfähigkeit zugesprochen werden. Mit hinreichend viel Papier wäre man also in der Lage, zu denken, ohne dabei denken zu müssen – nur durch das Befolgen syntaktischer Regeln. Denkbewegungen werden dabei schriftlich festgehalten und der Gang der Gedanken kann entsprechend vor Augen geführt und später nachgelesen werden. Es wäre auch möglich, jenen Teil des Menschen, an dem sich das Denken vollzieht, gemäß den Notationen von Standardbeschreibung und Beschreibungszahl als numerischen Wert anzugeben. Alle diese Umstände ergeben sich notwendigerweise aus der Annahme, dass Maschinen denken können, sowie der Annahme der Church-Turing-These. Demnach sind Mensch und Maschine sowohl hinsichtlich ihrer Rechenfähigkeiten als auch ihrer Denkfähigkeiten gleichmächtig. Im Folgenden werden zwei Einwände gegen die Gleichsetzung von Denken und Rechnen gegeben sowie Argumente gegen diese Einwände entwickelt.

Die viel langsamere Geschwindigkeit des rechnenden Denkens gegenüber dem vermeintlich menschlichen Denken ist ein möglicher Einwand

gegen die Zuschreibung der Denkfähigkeit. Man wäre beim Versuch, mit Schreiben und Radieren das Denken einer Turingmaschine zu simulieren, wohl wesentlich langsamer als beim eigenen Denken. Ich meine aber nicht, dass Denkgeschwindigkeit ein Kriterium für die Fähigkeit zu Denken ist. Wenn sie es wäre, dann wäre auch die Denkfähigkeit von Turingmaschinen zweifelhaft: In der Fassung von Turing (1936/37), steht ihnen unendlich viel Zeit für Operationen zur Verfügung. Dass auch Menschen sehr lange über ein Problem nachdenken können, rechtfertigt, dass eine Turingmaschine grenzenlos lange an einem Problem arbeiten kann. Turing (1996: 256) gibt eine Analogie dazu, wenn er schreibt, Mathematiker\_innen hätten hunderte Jahre über den Beweis des Satzes von Fermat, also über eine einzige Frage, nachgedacht. Mit dem Finden des Beweises im Jahr 1994 wurde nicht nur Fermats Satz – es gibt keine ganzzahligen Lösungen der Gleichung  $a^n + b^n = c^n$  für  $n > 2$  – bewiesen, es hat sich auch herausgestellt, dass die Frage nach dem Beweis eine entscheidbare Frage ist; auch eine Turingmaschine hätte die Lösung irgendwann gefunden.

Ein anderer möglicher Einwand ist, dass es sich beim besprochenen Begriff des Denkens nur um einen Sonderfall des menschlichen Denkens handelt – um ein regel- und zeichenorientiertes, arithmetisches Denken. Diesem Einwand kommt stärkend hinzu, dass Turing selbst ablehnt, die Frage zu behandeln, ob Maschinen denken können und es bevorzugt, von ‚machine intelligence‘ zu schreiben anstatt von ‚thinking machines‘ bzw., dass er den pragmatischen Ansatz des modifizierten Imitation-Game bevorzugt.

Gegen diesen Einwand kann die mögliche Übertragung des vermeintlich nicht regel- und zeichenorientierten Denkens in ein solches regel- und zeichenorientiertes Denken angeführt werden: Auch wenn wir phantasieren, kreativ sind oder dergleichen, laufen im Gehirn Vorgänge derselben Art ab wie beim Rechnen, ohne dass wir es wissen. Wenn Turing (1996: 257ff.) mögliche Lernverfahren für Maschinen beschreibt, auf eine Codierung für Lust- und Unlustsysteme eingeht und darüber nachdenkt, womit sich Maschinen die Zeit vertreiben würden – „mostly in playing games such as Chess and GO, and possibly Bridge“ (Turing 1996: 257) –, dann können diese Passagen so interpretiert werden, dass Turing an die Codierbarkeit aller möglichen Denkvorgänge glaubt. Auch die empirische Forschung hat an Turings Ideen Anstoß genommen: Der Neurophysiologe und Kybernetiker Warren S. McCulloch hat 1943 gemeinsam mit dem Logiker Walter H.

Pitts den Artikel *A Logical Calculus of the Ideas Immanent in Nervous Activity* im *Bulletin of Mathematical Biophysics* veröffentlicht. Darin entwickeln sie eine formallogische Notation für die verbundenen Neuronen des Gehirns. Wie McCulloch später bekannt gab, war ihre Arbeit wesentlich vom Konzept der Turingmaschine beeinflusst (vgl. Petzold 2008: 331ff.).

Auch wenn empirische Gehirnforschung an Turing anschließt, ist es kein einfaches Unterfangen, Turings Vorstellung des Gehirns als naiv-materialistische Ansicht zu charakterisieren. Er interessiert sich für die abstrakte und logische Modellierung von Denkvorgängen, nicht für deren empirische Abläufe. Die Grundlage für seine Vorstellung des menschlichen Gehirns waren also nicht biochemische, organische Vorgänge, sondern deren Abstraktion zu einem Modell von diskreten Zuständen und Übergängen zwischen diesen Zuständen. Es gibt auch Textstellen, die Zweifel daran aufkommen lassen, ob Turing überhaupt eine vollständige, materielle Erklärbarkeit des Menschen annimmt, etwa das Telepathie-Argument gegen das Imitation-Game (vgl. Turing 1950: 453f.). Dessen ungeachtet, gibt es genug Textstellen, die darauf hindeuten, dass Turing die Church-Turing-These bejaht, ohne sie beweisen zu können, und an die Simulation aller Denkvorgänge in einer Turingmaschine glaubt. Es ist zwar unklar, wie eine Turingmaschine aussieht, ob sie überhaupt irgendwie aussehen kann, es werden aber dennoch bildhafte Beschreibungen gegeben. Ich habe eine Schreibmaschine, die ihr Band selbst beschreibt, als Bild für die Turingmaschine erwähnt. Petzold (2008: 71) gibt ein anderes Bild dafür und schreibt: „What does a Turing machine look like? You can certainly imagine some crazy looking machine, but a better approach is to look in a mirror.“

## DENKENDE MENSCHEN

Im ersten Abschnitt habe ich einen Begriff des Denkens nach dem Vorbild der Turingmaschine beschrieben und gezeigt, welche Folgen aus einer Zuschreibung der Denkfähigkeit an Maschinen aufgrund der Äquivalenz zwischen verschiedenen Modellen für Berechenbarkeit sowie der Church-Turing-These entstehen. Im folgenden Abschnitt möchte ich drei Beispiele geben, bei denen ein nach dem Vorbild der Maschine verstandener Begriff des Denkens vorauselend als Erklärung des menschlichen Denkens herangezogen wird. Es geht also erneut um denselben Begriff des Denkens, dem

ich mich aber nicht von maschineller Seite, sondern von menschlicher Seite her annäherte.

1) Um online unerwünschte Einträge von Computerprogrammen zum Beispiel in Kommentarbereiche zu verhindern, müssen häufig Eingaben bei sogenannten CAPTCHA-Kontrollen gemacht werden. CAPTCHA ist ein Akronym von ‚Completely automated public Turing test to tell computers and humans apart‘ und bezeichnet solche Aufgaben, die für Menschen einfach und für Computer schwierig zu lösen sein sollten, wie etwa verzerrte Linien als Buchstaben zu identifizieren oder eine Frage in natürlicher Sprache zu beantworten. Manche dieser CAPTCHA-Kontrollen verhindern unerwünschte Einträge gut, andere weniger, manchmal ist auch für Menschen die richtige Lösung nicht so einfach zu erkennen. Besonders merkwürdig sind CAPTCHA-Kontrollen, die auf arithmetische Operationen zurückgreifen, und gesteigert wird die Merkwürdigkeit durch angegebene Fragen und Anweisungen, wie etwa: *Are you a human?  $3 + 4 = \dots$*

Die Lösung dieser Frage bedarf Intelligenz. Die Fähigkeit, eine solche Aufgabe zu lösen, kann allerdings nicht dafür eingesetzt werden, um menschliche von nicht-menschlichen Texteingaben zu unterscheiden. Eine solche Frage basiert implizit auf einem naiven Verständnis menschlicher Intelligenz, die nach dem Vorbild der Maschine verstanden wird, die selbst wiederum naiv gedacht wird. Gerade arithmetische Operationen sind das Metier des Computers. Ein Programm, das Text in Formularfelder einfügen kann, ist in der Regel auch dazu imstande, eine solche Aufgabe zu lösen. Auch die anderen erwähnten Aufgaben fallen in die Bereiche der klassischen KI-Forschung, sind also prototypische Maschinenprobleme: Mustererkennung und Verarbeitung natürlicher Sprache. Bereits in den 60er-Jahren des 20. Jahrhunderts waren Computerprogramme in der Lage, sogenannte eingekleidete Rechenaufgaben zu lösen. Etwa erwähnt Joseph Weizenbaum in *Computer Power and Human Reason* (1976) das Programm STUDENT von Daniel G. Bobrow aus dem Jahr 1967, das zum Beispiel auf folgende Frage in natürlicher Sprache die richtigen Antworten liefert: „Tom has twice as many fish as Mary has guppies. If Mary has 3 guppies, what is the number of fish Tom has?“ (Weizenbaum 1984: 187).

2) Ein anderes Beispiel stammt von Joseph Weizenbaum, der in erwähntem Buch über das Verhältnis von Intelligenz zum Intelligenzquotienten (IQ) schreibt. Er kritisiert ein Verständnis von Intelligenz als das, was bei Intelligenztests in der Einheit IQ gemessen wird. Er ist der Ansicht,

„that an entirely too simplistic notion of intelligence has dominated both popular and scientific thought, and that this notion is, in part, responsible for permitting artificial intelligence's perverse grand fantasy to grow“ (Weizenbaum 1984: 203). An mehreren Stellen fordert Weizenbaum dazu auf, Begriffe nicht losgelöst von ihren Bezugssystemen zu verstehen. Diese Forderung trifft heute auch auf sein Buch zu, das im Kontext der USA der 70er-Jahre zu lesen ist. So betrifft seine Kritik am IQ mitunter auch den Einfluss von Intelligenztests auf den Verlauf akademischer Karrieren und dergleichen (vgl. Weizenbaum 1984: 203f.). Seit der Veröffentlichung des Buchs vor rund 40 Jahren haben sich zwar einige Kritikpunkte geändert, der Kern von Weizenbaums Ansichten ist meines Erachtens auch heute noch gültig.

Weizenbaums Kritikpunkt ist, dass sich menschliche Intelligenz weder mit den eingesetzten Intelligenztests messen lässt, noch vom IQ akkurat erfasst wird. Nach Weizenbaum lasse sich menschliche Intelligenz nämlich nicht linear messen, sie sei verschränkt mit wesentlich menschlichen Charakteristika wie Kreativität, Intuition und Klugheit. Es wäre lächerlich, eine lineare Skala für Kreativität und dergleichen zu erarbeiten. Intelligenz kann auch nicht ohne Bezugssystem verstanden werden: Man kann intelligent handeln, intelligent denken, intelligent urteilen etc. Um Intelligenz zu messen, müsse sie isoliert als intelligente Intelligenz verstanden werden, was ein unsinniges Maß wäre. (Vgl. ebd.: 204f.)

Nach Weizenbaum ist die Gleichsetzung von Intelligenz mit dem IQ wesentlich verantwortlich für das Entstehen von Ansichten, nach denen es keinen Bereich des menschlichen Denkens gebe, der nicht auch maschinell erfasst werden kann (vgl. ebd.: 207). Ein intelligenter Mensch hat Begabung dafür, geschickt Probleme eines bestimmten Bereichs zu lösen – das kann etwa Wissenschaft, Alltag, Politik oder Kultur im Allgemeinen betreffen. Die Entwicklung dieser Begabung ist, was Weizenbaum als Sozialisierung versteht: „Every organism is socialized by the process of dealing with problems that confront it.“ (Ebd.: 210) Was das menschliche Denken bestimmt, ist, dass es sich mit Problemen des Menschen beschäftigt. Diese Beschäftigung ist nicht alleine von Logik geleitet, sondern auch von Ängsten, Hoffnungen, Wünschen etc., das heißt, die Beschäftigung mit menschlichen Problemen macht Menschen zu Menschen – formale Probleme sind ein Teil der menschlichen Probleme, aber nicht ihre Gesamtheit.

3) Im dritten Beispiel möchte ich mich auf Verhaltensforschung beziehen, bei der untersucht wird, ob und wie Tiere denken können. Es gibt einen methodischen und einen sprachlichen Aspekt, den ich erwähnen möchte: Der methodische Aspekt betrifft den Aufbau bestimmter Experimente. Der experimentelle Nachweis von kognitiven Fähigkeiten erfolgt häufig innerhalb einer Experimentumwelt und gemäß einer Denkfähigkeit nach maschinellem Vorbild: Tiere lösen logische Rätsel, müssen mit Zeichen auf Zeichen antworten, Gegenstände nach bestimmten Eigenschaften anordnen etc. Solche Aufgaben fallen ebenso in die klassischen Bereiche der KI-Forschung: Logik, Sprachverarbeitung und Mustererkennung.

Ein Beispiel dafür ist die Primatenforschung von Nobuyuki Kawai und Tetsuro Matsuzawa. Sie berichten in der Zeitschrift *Nature* von einem Experiment mit dem Schimpanseweibchen Ai, das die arabischen Ziffern von null bis neun der Größe nach ordnen kann. Im beschriebenen Experiment konnte nachgewiesen werden, dass Ai bestimmte Folgen von fünf Ziffern im Gedächtnis behalten konnte. Ihre Leistung wird mit den Fähigkeiten von menschlichen Kindern vor der Schulzeit verglichen. (Vgl. Kawai & Matsuzawa 2000: 39)

Der sprachliche Aspekt betrifft Beschreibungen von Tierverhalten. Solche Beschreibungen sind nicht einfach: Sätze wie *Die Katze weiß, wann sie kommt* und dergleichen sind im Alltag nicht problematisch, im philosophischen Gebrauch handelt es sich allerdings um eine Wissenszuschreibung, die Voraussetzungen und Konsequenzen hat. In der Verhaltensforschung ist man sich der Schwierigkeit der angemessenen Wortwahl durchaus bewusst, sie wird etwa von Kuczaj II und Tahmes Walker (2006: 582) thematisiert. Das von mir angesprochene Problem betrifft aber gar nicht einen sorglosen Umgang mit philosophisch aufgeladenen Begriffen außerhalb der Philosophie, sondern die Verwendung einer Sprache, die von technischen Terminen durchsetzt ist. Die folgenden Beispiele sind aus dem Textzusammenhang losgelöst, aber – provokant formuliert – bei ihnen ist es unklar, ob die Bezeichnung einer Tiergattung verwendet wird oder der Eigenname einer bestimmten Maschine: „Dolphin processing of gestural sequences“ (Kuczaj II & Tahmes Walker 2006: 582); „[...] hoverflies appear to compute their interception paths with conspecifics according to a simply specifiable rule, one which could plausibly be ‚hard-wired‘ into the flies‘ brains [...]“ (Boden 1984: 155); „[...] continuing debate regarding the nature of the stored

information used by pigeons during complex visual categorizations [...]“ (Huber & Aust 2006: 328).

Jedes der drei Beispiele kann als Symptom dafür verstanden werden, dass ein Begriff des Denkens nach Vorbild der Maschine zur Anwendung kommt, wenn es um eine Zuschreibung von Denkfähigkeit geht. Die Beispiele veranschaulichen, dass eine für Menschen und Maschinen analoge Vorstellung des Denkens auch auf andere Bereiche als Rechenfähigkeit ausgedehnt wird: Als Mittel zur Unterscheidung zwischen Mensch und Maschine (erstes Beispiel) sowie als Maßstab für die Intelligenz von Menschen (zweites Beispiel) und von Tieren (drittes Beispiel). Demnach müsste die eingangs von Joseph M. Rosen zitierte Enttäuschung, dass es sich bei simulierten Denkvorgängen nicht um ein wirkliches Denken handelt, verschwinden. Maschinen kann die Fähigkeit zu denken zugeschrieben werden, wenn man andere mögliche Interpretationen des Denkens aus den Augen verliert. Man überwindet damit einen diffusen Begriff des Denkens zugunsten einer begrifflichen Festlegung.

Eine Annäherung des Menschen an die Maschine wird angesichts neuer Technologien zunehmend deutlich. Viele Aspekte dieser Annäherung betreffen den menschlichen Körper, der durch technische Eingriffe verbessert und verschönert sowie von Krankheiten und Verletzungen geheilt werden kann. Diskurse darüber verlaufen zwischen zwei romantischen Fiktionen: jener des Körpers als unantastbare Natur und jener des Körpers als posthumane Informationsgemenge. Hinsichtlich des Denkens scheinen Maschine und Mensch ihre Rollen zu tauschen: Das menschliche Gehirn wird zum Datenträger reduziert, der Computerspeicher zum Gedankenträger erhoben. Denkende Maschinen hätten gegenüber denkenden Menschen zahlreiche Vorteile: Sie können zwar falsch programmiert sein – das wäre ein Fehler, der aus menschlichem Versagen resultiert –, aber sie können sich nicht irren, sie erkranken nicht, sie werden nicht müde, hungrig oder durstig, sie haben keine ablenkenden Interessen etc.

Was bei solchen Ansichten allerdings unerwähnt bleibt, ist, dass denkende Wesen nicht nur bestimmte, in der Regel positiv besetzte Rechte und Freiheiten besitzen, sondern auch Verantwortung und Pflichten haben: Denkende Menschen tragen politische Verantwortung, sie nehmen – wenn es möglich ist – an demokratischen Wahlen teil, sie gehen mit anderen denkenden Menschen sexuelle, das heißt körperliche Beziehungen ein, sie diskutieren über Entscheidungen, die für sie und andere mitunter lebensent-

scheidend sind, sie treffen solche Entscheidungen. Denkende Menschen töten sich selbst und gegenseitig, sie helfen einander, und beides nicht aus einem einzigen bestimmten Trieb oder einer fassbaren Menge von Regeln heraus, sondern aus einer Vielzahl an Gründen.

Es ist sinnvoll, Maschinen – und auch Tiere – von diesen Verpflichtungen auszuschließen. Ein guter Grund für diesen Ausschluss ist, anzunehmen, dass sie nicht in der Lage sind, so zu denken wie Menschen denken. Wie kann ein Computer dazu in der Lage sein, Gerichtsurteile zu fällen? Wie kann ein Computer eine medizinische Diagnose erstellen und Patient\_innen über ernste Erkrankungen informieren? Wie kann ein Computer politische Entscheidungen treffen? Selbst wenn man die abwegige Annahme trifft, solche Entscheidungsfindungsprozesse ließen sich modellieren, ist es absurd anzunehmen, dass eine Maschine verstehen oder simulieren könne, was es heißt, mit einer getroffenen Entscheidung zu leben. Ihr Leben – wenn man von Leben sprechen kann – besteht eben nicht darin zu irren, zu erkranken, müde zu werden, zu hungrig etc. Gedankenexperimente über Totalsimulationen, in denen auch für Maschinen die Vorstellung eines mit der Umwelt verbundenen Körpers erklärt wird, sind interessante Überlegungen, die Ausführungen hier sind aber nicht aus hypothetischen, sondern empirischen Beispielen hervorgegangen.

Aus meinen Überlegungen lassen sich keine ethischen Schlüsse ziehen: Maschinen – und auch Tieren – keine Denkfähigkeit zuzusprechen, rechtfertigt weder Grausamkeit an ihnen noch Respekt für sie. Ihnen Denkfähigkeit zuzusprechen, lässt schwierige Fragen auftreten: Welche Art von Verantwortung wird ihnen wie und warum zugeschrieben? Was soll mit fehlerhaften Computern passieren? Soll ihr Speicher gelöscht oder behandelt werden? Es handelt sich dabei um den materiellen Träger von Gedanken. Es gäbe nur fadenscheinige Gründe, fehlerhafte, denkfähige Computer anders zu behandeln als kranke, denkfähige Menschen. Die Vorstellung eines Krankenhauses, in dem neurotische Computer behandelt werden, scheint seltsam. Wird Maschinen – und Tieren – Denkfähigkeit nicht zugesprochen, entstehen diese Fragen zwar auch, die Stellungnahme dazu hat aber andere Auswirkungen. Werden diese Fragen ohne zugeschriebene Denkfähigkeit behandelt, münden sie in der sinnvollen, ethischen Frage, welche Stellung Maschinen und Tiere in der Welt des Menschen einnehmen.

## RESÜMEE

Dieser Text ist aus Überlegungen darüber entstanden, welcher Begriff des Denkens sich aus Betrachtungen von Maschinen bilden lässt und was passiert, wenn dieses Modell auf menschliches Denken angewandt wird. Die Äquivalenz bestimmter formaler Modelle von Berechenbarkeit kann gezeigt werden, das heißt mit einem Modell  $M_1$  lassen sich nicht mehr und nicht weniger Probleme lösen als mit einem anderen, äquivalenten Modell  $M_2$ . Es sind keine Modelle bekannt, die mehr leisten können als die Turingmaschine und zu ihr äquivalente Konzepte. Dass auch Menschen hinsichtlich ihrer Rechenfähigkeiten nicht mehr leisten können als Turingmaschinen, ist der Inhalt der Church-Turing-These. Diese These kann nicht bewiesen werden, im Allgemeinen wird aber davon ausgegangen, dass sie wahr ist.

Computer haben bereits Schachpartien gegen amtierende Schachweltmeister gewonnen und wenn Autos sich in Zukunft selbst fahren, werden sie ohne Zweifel besser einparken können als Menschen. Angesichts der erstaunlichen Fähigkeiten von Maschinen kommen Zweifel daran auf, ob sie nicht doch gleich denkfähig wie Menschen sind und hinsichtlich ihrer Eigenschaften – endlose Rationalität ohne Erschöpfung – nicht sogar buchstäblich bessere Menschen wären.

Meine Antwort darauf ist, dass solche Überlegungen darauf hinarbeiten, die Verantwortung für sich selbst und andere Menschen aufzugeben. Diese Verantwortung entsteht aus menschlicher Sozialisierung und sie zu tragen ist nicht einfach, aber wesentlicher Bestandteil des anstrengenden menschlichen Lebens. In Anbetracht der vielen Konflikte, mit denen man konfrontiert wird – seien sie alltäglicher, sozialer, ethischer, ästhetischer, emotionaler, rationaler etc. Art –, sind jene Probleme, für die nur Rationalität benötigt wird, ein kleiner Teilbereich. Ich möchte mich nicht für bedingungslose Irrationalität in diesen Bereichen aussprechen, aber verneine die pauschale Möglichkeit von geeigneten Konfliktmodellierungen. Um Entscheidungen in solchen Konflikten zu treffen, bedarf es nicht nur der Fähigkeit, rational zu denken, sondern auch Vorstellungen davon, was es heißt, zu leben, und davon, wie andere Menschen angesichts bestimmter Umstände zu leben haben. Ich möchte also den Unterschied zwischen menschlichem Denken und maschinellem Denken bejahen – nicht aufgrund des mit ihm verbundenen Ausmaßes an Verantwortung. Der Fähigkeit zu rechnen kommt keine Ver-

antwortung in dem Sinn zu, wie sie der Fähigkeit, Entscheidungen in menschlichen Konflikten zu treffen, zukommt. Es erscheint mir müßig, jemanden davon überzeugen zu wollen, dass eine Maschine verstehen könne, was es heißt, mit bestimmten Entscheidungen leben zu müssen. Die Fähigkeit zu denken auch Maschinen zuzusprechen, bedeutet, Verantwortung abzugeben, und zeugt von einer Flucht des Menschen vor sich selbst.

## LITERATUR

- Boden, M.A. (1984): Animal Perception from an Artificial Intelligence Viewpoint, in: Hookway, Ch. (Ed.), *Minds, Machines and Evolution. Philosophical studies*. Cambridge University Press, 153–174.
- Church, A. (1937): Reviewed Work: On Computable Numbers, with an Application to the Entscheidungsproblem by A. M. Turing, in: *The Journal of Symbolic Logic*. 2 (1), 42–43.
- Clarke, A.C. (2000): *2001: A Space Odyssey*. New York: ROC.
- Herold, H., Lurz, B., Wohlrab, J. (2006): *Grundlagen der Informatik*. München: Pearson Studium.
- Hodges, A. (2014): *Alan Turing: The enigma*. London: Vintage.
- Huber, L., Aust, U. (2006): A Modified Feature Theory as an Account of Pigeon Visual Categorization, in: Wasserman, E.A., Zentall, Th.R. (Ed.), *Comparative Cognition. Experimental Explorations of Animal Intelligence*. New York: Oxford University Press, 325–342.
- Hopcroft, J.E., Motwani, R., Ullman, J.D. (2002): *Einführung in die Automatentheorie, formale Sprachen und Komplexitätstheorie*. München: Pearson Studium.
- Kawai, N., Matsuzawa, T. (2000): Numerical Memory Span in a Chimpanzee, in: *Nature*. 403 (6765), 39–40.
- Kleene, S.C. (1967): *Mathematical Logic*. New York: John Wiley & Sons.
- Kleene, S.C. (1971): *Introduction to Metamathematics*. Groningen, Amsterdam: Wolters-Noordhoff Publishing & North-Holland Publishing.
- Kuczaj II, S.A., Tahmes Walker, R. (2006): How Do Dolphins Solve Problems? in: Wasserman, E.A., Zentall, Th.R. (Ed.), *Comparative Cognition. Experimental Explorations of Animal Intelligence*. New York: Oxford University Press, 580–601.

- Neumann, J. von. (1992): *First Draft of a Report on the EDVAC*. Corrected copy of the original typescript by Michael D. Godfrey, Information Systems Laboratory, Stanford University. Retrieved from: [http://www.wiley.com/legacy/wileychi/wang\\_archi/supp/appendix\\_a.pdf](http://www.wiley.com/legacy/wileychi/wang_archi/supp/appendix_a.pdf) [25.10.2015].
- Oppy, G., Dowe, D. (2011): The Turing Test, in: Zalta, E.N. (Ed.), *The Stanford Encyclopedia of Philosophy (Spring 2011 Edition)*. Retrieved from: <http://plato.stanford.edu/archives/spr2011/entries/turing-test> [25.10.2015].
- Petzold, C. (2008): *The annotated Turing. A Guided Tour through Alan Turing's Historic Paper on Computability and the Turing Machine*. Wiley Publishing: Indianapolis.
- Turing, A.M. (o.J.): *Proposed Electronic Calculator*. Retrieved from: [http://www.alanturing.net/turing\\_archive/archive/p/p01/p01.php](http://www.alanturing.net/turing_archive/archive/p/p01/p01.php) [25.10.2015].
- Turing, A.M. (1936/37): On Computable Numbers, with an Application to the Entscheidungsproblem, in: *Proceedings of the London Mathematical Society*. 42 (3/4), 230–265.
- Turing, A.M. (1950): Machinery and Intelligence, in: *Mind*. 59, 433–460.
- Turing, A.M. (1996): Intelligent Machinery, A Heretical Theory, in: *Philosophia Mathematica*. 4 (3), 256–260.
- Warwick, K., Shah, H. (2015): Can Machines Think? A Report on Turing Test Experiments at the Royal Society, in: *Journal of Experimental & Theoretical Artificial Intelligence*. Volume in press. Retrieved from: <http://tandfonline.com/doi/full/10.1080/0952813X.2015.1055826> [25.10.2015].
- Weizenbaum, J. (1984): *Computer Power and Human Reason. From Judgement to Calculation*. Harmondsworth: Penguin Books.

## FILME

- TECHNOCALYPS (2006): R: Frank Theys. Belgien: Votnik. Retrieved from: <https://www.youtube.com/watch?v=7MXQSbjBL7Q> [25.10.2015]

