UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

SOBRE EL TEST DE TURING: LA MÁQUINA COMO INTERROGADORA

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

MATEMÁTICA

P R E S E N T A :

ANA KAREN FLORES GARCÍA

DIRECTOR DE TESIS:
DR. CARLOS TORRES ALCARAZ
2017

CIUDAD UNIVERSITARIA, CDMX

Índice general

Introducción

En Octubre de 1950 el matemático inglés Alan M. Turing publicó un artículo titulado *Computing Machinery and Intelligence*, en el cual intenta responder formalmente al cuestionamiento sobre si las máquinas pueden pensar. El efecto que tuvo dicho artículo en la comunidad científica, filosófica y psicológica se puede percibir incluso en la actualidad, a más de 60 años de haber sido publicado.

Turing plantea en su artículo que la pregunta acerca de la posibilidad de que las máquinas piensen puede ser reemplazada por un problema de decisión que llama *Juego de la Imitación* y que más tarde es conocido como *Test de Turing*. Dicho test está basado en un juego que prueba la capacidad de un interrogador humano para diferenciar a un hombre de una mujer por medio de una conversación escrita. En este juego el hombre tiene como objetivo engañar al interrogador haciéndole creer que él es la mujer. El test plantea que si una máquina reemplaza al hombre en dicha conversación logrando que el interrogador se equivoque una cantidad de veces igual a la que se equivocaría si estuviera conversando con el humano, entonces podemos decir que la máquina *piensa*.

Desde el momento de su publicación hasta hoy en día, el test de Turing ha recibido tantas críticas positivas como negativas. Uno de los objetivos de este trabajo es analizar las críticas que, en mi opinión, son más relevantes, y exponer las razones por las cuáles creo que el test de Turing es la prueba más acertada que tenemos a la fecha de pensamiento inteligente.

Dentro de las críticas hay algunas que intentan minimizar el valor del test de Turing. Aún así, la influencia directa que éste ha tenido sobre varios campos del conocimiento, es innegable. En psicología, por ejemplo, ha servido principalmente para el análisis y entendimiento de la naturaleza del pensamiento humano y más aún, para el estudio de los mecanismos usados por los humanos para adjudicar pensamiento inteligente a terceros.

La capacidad de reconocer y adjudicar dichos procesos a nosotros mismos y a terceros, se conoce como "Teoría de la Mente", y es el estudio de ésta lo que generó el planteamiento

de nuevas versiones del test de Turing. La versión que se propone en esta tesis está basada en la consideración del interrogador como factor fundamental del éxito o fracaso del test y, su objetivo es diferenciar humanos de máquinas y no hombres de mujeres.

Por mucho tiempo se creyó que la máquina *imitadora* desempeñaba el papel más importante en el test, dejando a un lado el estudio del interrogador como agente activo. Debido a los estudios mencionados anteriormente acerca de la teoría de la mente, el papel del interrogador obtuvo solidez, permitiendo modelar nuevas versiones del test. Nuestra versión intenta poner a prueba esta vez la capacidad de otra máquina para diferenciar a un humano de una máquina por medio de una conversación, es decir, la capacidad de una máquina para jugar como interrogador en el test.

Ésta nueva interpretación deja de tener por objetivo responder a la cuestión de la existencia de una máquina que logre engañar a un humano haciéndole creer que es un humano (la mayoría de las veces), para enfocarse en la posible existencia de una máquina que siempre logre distinguir entre máquinas y humanos.

Los profesores e investigadores japoneses Yuzuro Sato y Takashi Ikegami, intentaron probar en dos artículos publicados en 1999 y 2004[1], que la existencia de una máquina como la que describimos es imposible. Lo que intentan es modelar una demostración que usa un método muy común en teoría de la computación llamado *diagonalización*. Éste suele usarse para demostrar la inexistencia de sistemas que llamamos *perfectos*.

Otro de los objetivos de esta tesis, es examinar las demostraciones dadas por Sato e Ikegami, mostrando que son incorrectas. Lo que hacemos es exhibir las razones por las cuales son erróneas y exponer una nueva demostración, en donde se prueba correctamente la imposibilidad de la existencia de dicho interrogador.

Éste resultado, además de implicar ciertas limitaciones en las máquinas de Turing, se usa como ejemplo para evidenciar el alcance del test al proporcionar valiosa información acerca del pensamiento humano y el de las máquinas.

Con base en lo anterior, se plantea una tesis que consta de cuatro capítulos.

Capítulo 1. En este espacio se introduce el *test de Turing*, tal como lo planteara Turing en su artículo de 1950[2]. Además, se hace un breve análisis del artículo exponiendo los argumentos a favor y en contra que, para mí, son fundamentales en la discusión sobre el pensamiento de las máquinas.

[1] Sato, Yusuru & Ikegami, Takashi, 2004 & 1999, "Undecidability in the Imitation Game", Minds and Machines, 14: 133-143

[2] Turing, A. M. (1950) "Computing machinery and intelligence", Mind LIX(2236):433-460.

Capítulo 2. En este lugar se analiza la manera en que los humanos solemos atribuir pensamiento inteligente a terceros y la forma en que lo hacemos, todo con relación al test de Turing. También se explica una nueva versión del test, en la que tanto el interrogador como el imitador son máquinas de Turing.

Capítulo 3. En este lugar se plantea formalmente la nueva versión del test de Turing y se explica el teorema a demostrar con base en ésta. Además, se exponen las razones por las cuales las demostraciones dadas por Sato e Ikegami del teorema son erróneas. Por último, se ofrece una demostración correcta del teorema.

Capítulo 4. En este lugar se examinan los alcances del test de Turing con base en el resultado anterior, no sólo con relación al reconocimiento del pensamiento humano, sino el de las máquinas. Además, se hace un último análisis comparando el test de Turing con otras pruebas de inteligencia, con el fin de exhibir el valor e importancia de dicho test. Se concluye con una reflexión respecto a las implicaciones que el resultado recién demostrado podría tener, por ejemplo, en relación a la tesis de Church-Turing.

Capítulo 1

El test de Turing

¿Pueden pensar las máquinas? Con esta pregunta Alan Turing en su artículo de 1950 abre una incógnita que ha perdurado hasta nuestros días. Responder a esta pregunta sin convenciones de lo que entendemos por "pensar" y por "máquina", resulta no sólo complejo sino totalmente inútil, puesto que para cada individuo estas dos palabras pueden tener distintos significados y valores. Para intentar responder a esto de una manera más concreta, Turing hizo dos cosas: primero, propuso un modelo matemático (desde 1936) de lo que es una máquina; segundo, trasladó la cuestión de inteligencia a una prueba conocida como "test de Turing", la cual tiene como base un juego de salón conocido como "juego de la imitación", en el que participan 3 jugadores: un hombre (A), una mujer (B) y un interrogador (C) (que puede ser tanto hombre como mujer). C se encuentra separado de los otros dos jugadores y no puede verlos ni escucharlos, sólo los conoce como X e Y respectivamente; además, la forma de comunicación es escrita (idealmente por medio de un teletipo) con el propósito de evitar el reconocimiento caligráfico. El objetivo del juego para C es averiguar quién es el hombre y quién es la mujer; el del hombre (A) inducir al interrogador a hacer una identificación errónea; y el de la mujer (B) es colaborar para que el interrogador haga la identificación correcta.

El interrogador puede hacer preguntas del tipo: *¿puede decirme X de qué largo tiene el cabello?* (Turing, 1950), a las que los otros dos jugadores podrán contestar de la manera mas conveniente y convincente para lograr su cometido; por lo demás, el interrogador no puede pedir ningún tipo de manifestación práctica a los competidores (v.gr. que le envíe una muestra de su cabello). Estas son las reglas del juego, dentro de las cuales nunca se habla de límite de tiempo ni de ninguna otra restricción.

Ya explicado el procedimiento del juego, Turing plantea las siguientes cuestiones: "¿qué sucederá cuando una máquina tome el rol del jugador A?, ¿decidirá equivocadamente

el interrogador con la misma frecuencia cuando juega con un hombre que cuando juega con una máquina?" (Turing, 1950).

Según Turing, estos cuestionamientos reemplazan efectivamente la pregunta principal acerca del pensamiento de las máquinas, formulando entonces el siguiente planteamiento: si para el juego de la imitación existe una máquina que logre engañar al interrogador haciéndole creer que es una mujer o que el otro jugador es un hombre, una cantidad de veces equivalente a la que ocurriría si el juego se diera entre humanos, podría decirse que dicha máquina *piensa* y por lo tanto que las máquinas pueden pensar.[1]

Hay que aclarar que Turing no profundiza en el hecho de que una máquina pase el test o no, así como tampoco se pregunta desde un inicio, por ejemplo, qué podríamos deducir de un hombre que en el juego de la imitación logre engañar al interrogador haciéndole creer que es una mujer.

Douglas Hofstadter (Hofstadter 1981) plantea que lo que podríamos inferir de que un hombre gane en el juego de la imitación es que tiene buena intuición de lo que es la *mentalidad femenina* y, por lo tanto, que lo que podemos inferir de que una máquina gane en el juego de la imitación es que tiene buena intuición en lo que respecta a ser humano, es decir, que puede perfectamente simular el pensamiento humano.

Ahora bien, ¿por qué según Turing, el que la máquina engañe por escrito es suficientemente válido para decir que piensa y no por ejemplo el hecho de que logre bailar tan bien que no parezca una máquina? Intentaré aclarar este punto empezando por lo siguiente: un valioso logro de Turing, es que establece una línea divisoria bastante clara entre las capacidades físicas y las capacidades intelectuales del ser humano, para así poder delimitar las características del pensamiento y poder identificar qué tipo de evidencia necesitamos para poder afirmar su existencia.

Es claro que si convenimos que alguna capacidad física implica la existencia del pensamiento, podríamos entonces dar como ejemplo de ser pensante a cualquiera de los robots que existen actualmente que pueden bailar o jugar futbol, a lo cual seguramente seguirían infinidad de réplicas en contra de dicha afirmación. De la misma manera, al querer que la máquina gane el juego de la imitación no pretendemos que al interrogador se le permita hacer peticiones de evidencia física, con el fin de no caer en situaciones injustas como el culpar a la máquina por no ganar en un concurso de belleza (Turing, 1950).

[1] A pesar de la relación directa que podemos encontrar entre el juego de la imitación original y aquel modificado por Turing, me parece pertinente aclarar que en lo que sigue, el test se entiende (incluso por Turing) como una prueba entre un interrogador humano y una máquina, la cual tiene por objetivo la imitación de un ser humano y no de una mujer específicamente.

Por lo tanto, parece sensato limitarnos a un método de decisión de tipo intelectual como, por ejemplo, el método de pregunta y respuesta que ofrece el test de Turing, el cual logra abarcar la mayor cantidad de campos de desempeño humano que nos interesan (Turing, 1950).

Al igual que en varios tópicos de su artículo de 1950, Turing es ambiguo en su manera de responder a ciertas preguntas (por ejemplo qué opina del éxito o del fracaso de su test). El por qué el test es una prueba válida para la inteligencia de las máquinas es una de ellas. Sin embargo, deja claro que para él, si una máquina llegara a aprobar el test ganando el juego de la imitación con cierta regularidad, es porque dicha máquina es capaz de realizar actos que se consideran inteligentes. Esto es producto de la confianza que tiene en la correcta demarcación que hace el test respecto a las capacidades físicas e intelectuales del ser humano; creo además que esto se debe también a la época en que Turing diseña su test, en la que un tema central era investigar la estructura y el funcionamiento de la mente humana. Para Turing, tanto la capacidad de fingir o engañar, eran signos de inteligencia en el humano [2]. Aunado a esto, Turing da un claro ejemplo de la fuerza de su test al replicar a la objeción de la consciencia, según la cual no es suficiente que una máquina pueda tener una conversación para que podamos decir que piensa, más bien, lo que necesitamos es que sea consciente de lo que está diciendo, es decir, que sepa el significado de sus palabras. El ejemplo es el siguiente:

Interrogador: En la primera línea de tu soneto que dice "Podría compararte con un día de verano", ¿no quedaría igual de bien o mejor decir "con un día de primavera?".

testigo: No, no quedaría.

Interrogador: ¿Y qué tal con "un día de invierno"? Eso quedaría perfecto.

testigo: Sí, pero nadie quiere ser comparado con un día de invierno.

Interrogador: ¿Dirías que el Sr. Pickwick te hizo recordar Navidad?

testigo: De alguna manera, sí.

Interrogador: Además Navidad es un día de invierno y no creo que al Sr. Pickwick le importe la comparación.

testigo: ¿Es una broma? Por un día de invierno uno se refiere a un típico día de invierno no a un día especial como Navidad.

[2]Darwin, C. (1877) "A biographical sketch of an infant mind", Mind

Y sigue así. El testigo claramente no sólo demuestra un entendimiento del idioma español sino de poesía, de las estaciones del año, de los sentimientos de la gente, etc. En palabras de Turing, si logramos que una máquina responda como este testigo, ¿seguiríamos afirmando que la máquina no sabe de lo que está hablando?

Con esto, Turing nos exhorta a aceptar el test como una prueba válida para la inteligencia de las máquinas y a juntar esfuerzos para la creación de una que lo apruebe.

Aclaremos que el que una máquina falle en el test no implica la falta de inteligencia en ella. Un humano inteligente podría no ganar en el juego de la imitación y por lo tanto parece justo permitir a las máquinas la misma situación. Esto es entonces, un test con una sola dirección, fallarlo no es prueba de nada.

Además, notemos que Turing no estaba comprometido con la idea de que pensar significa pensar como ser humano, no más que con la idea de que para que un hombre piense tiene que pensar como mujer. Hombres, mujeres y máquinas pueden tener distintas maneras de pensar. Pero seguramente, decía Turing, si alguno de ellos puede lograr con su propio pensamiento imitar a un hombre o mujer pensante, puede pensar en sí (Dennett 1985).

En el mismo artículo, Turing enfrenta varias de las objeciones que su test recibió, de las cuales analizaré brevemente las que creo se ajustan más al objetivo de esta tesis.

La primer objeción es la objeción teológica, según la cual la función de pensamiento es dada por Dios sólo a las almas inmortales y, dado que sólo los humanos y no los animales o las máquinas son portadores de almas inmortales, entonces sólo los humanos pueden pensar. Turing responde a esto considerando dos cosas, la primera es que no podemos limitar la omnipotencia de Dios en caso de que desee otorgar un alma a una máquina o a un animal; la segunda es un llamado a recordar que existieron casos en los que la iglesia condenó y negó hechos científicos que resultaron verdaderos.

Otra de las objeciones afirma que el que las máquinas piensen podría provocar un riesgo a futuro. Turing considera esta objeción insubstancial, pues cree que dicho argumento tiene más fuerza en manos de gente intelectual, ya que le dan un valor más alto al pensamiento que el resto de la humanidad y por lo tanto, responde sólo con una nota de consuelo.

La objeción matemática se basa en el segundo teorema de incompletud de Gödel, el cual nos lleva a concluir que las máquinas tienen limitaciones y por lo tanto que habrán ciertos problemas que no podrán resolver. Turing responde que al igual que las máquinas, el ser humano puede no ser capaz de resolver algunos problemas, la cual no es un obstáculo para su desarrollo cognitivo.

La objeción de la consciencia es una de las más fuertes, pues además de afirmar la imposibilidad de las máquinas de hacer ciertas cosas, afirma la imposibilidad de la máquina de ser consciente de haber hecho (o no) esas cosas. Turing responde que la única manera de asegurarnos que el otro (ya sea humano o máquina) es consciente o no, es la de ser ese otro y por lo tanto, siguiendo esta línea, caeríamos en una idea solipsista del mundo. Así, afirma que la manera natural del ser humano de adjudicar pensamiento a un tercero es por medio del monitoreo de sus acciones y, por esta razón, el test que propone resulta ser una buena herramienta para dicho monitoreo.

La objeción de las varias incapacidades se basa (irónicamente) en la incapacidad del ser humano de poder atribuir a las máquinas ciertas características como la de ser amable, bella, tener iniciativa, enamorarse, etc., debido a la imposibilidad de crear un programa que logre tener dichos atributos. Turing cree que esta percepción que se tiene de las máquinas se debe al principio de inducción científica, por medio del cual el ser humano a lo largo de su existencia ha ido caracterizando a las máquinas conforme a su composición material, quedándose con la idea de que son aparatos metálicos, fríos y sin variedad de comportamiento; cree además, que esta manera inconsciente de proceder pueder ser contraproducente pues nos puede llevar a conclusiones no fiables. Pues de ser así, podríamos, por ejemplo, pensar que como la mayoría de las personas que conocemos hablan español, entonces todos hablan español y por lo tanto es inútil intentar aprender otro idioma.

Éstas y otras más, son las objeciones a las que Turing responde.

1.1. Seguimiento de argumentos a favor y en contra

En esta sección analizaré dos perspectivas opuestas del test de Turing. Me basaré en los argumentos a favor y en contra que dan los norteamericanos Douglas Hofstadter y John Searle, respectivamente, y que son para mí pilares en la discusión.

Empezaremos por exponer el argumento en contra que da el filósofo John Searle, conocido como “el argumento de la habitación china”. En términos generales, éste consiste en la reducción del supuesto pensamiento de las máquinas, a un proceso consistente en relacionar símbolos basado en reglas impartidas por un operador. Este proceso no incluiría la comprensión de sentidos y significados, y funcionaría bajo principios meramente sintácticos (Searle, 1983).

Concretamente, este argumento se basa en un experimento mental en el cual el propio Searle se imagina dentro de una habitación en la que no tiene contacto físico con nadie. Lo único que se le entrega por medio de una ventana es un escrito en idioma chino (que

no entiende en absoluto), y dos escritos más, uno en chino y otro en inglés (idioma que sí entiende). Este último contiene instrucciones que relacionan el primer y el segundo escrito en chino; a continuación se le entrega un par más de escritos, uno en chino y el otro en inglés, que al igual contiene instrucciones que relacionan los escritos en chino anteriores con el tercer escrito (la relación se basa en la forma sintáctica y no en el significado).

Con toda esta documentación Searle es capaz de *aparentar* conocimiento en varios asuntos que le son consultados (todo por medio escrito y no físico), así como conocimiento del idioma chino (entregando un documento cuando se le presentan los escritos anteriores); es decir, relaciona símbolos que no conoce por medio de instrucciones que sí conoce, para así poder entablar una conversación con la gente fuera de la habitación, haciéndoles creer que conoce los temas que le son consultados y el idioma en el que le son consultados.

Veamos cómo esto se relaciona con el test de Turing. Las instrucciones dadas en inglés jugarían el papel del programa de la máquina. Estas instrucciones permitirían que un humano o una máquina pudiera correlacionar símbolos que no entiende, lo que para un observador externo (hablante del chino) sería una serie de preguntas y respuestas lógicas, coherentes y acertadas (todo esto en forma escrita). El observador o interrogador, al no tener acceso a la habitación y no poder ver el proceso con el que Searle construye sus escritos, no tiene forma de saber que éste, a pesar de dar respuestas acertadas, no tiene idea de qué está diciendo (pues sólo relacionó símbolos sin comprenderlos). Es más, si los agentes exteriores hicieran preguntas tanto en inglés como en chino y recibieran respuestas coherentes, podrían afirmar que el mismo proceso de pensamiento subyace a ambas (Barón, 2008).

El punto clave en el argumento de Searle en contra del pensamiento de las máquinas es la falta de *intencionalidad* de éste. La intencionalidad se ha considerado por mucho tiempo como característica fundamental de los procesos mentales del ser humano, consistiendo en el hecho de *estar dirigida a algo, ser acerca de algo* (Acero, 1995). Las relaciones simbólicas que hacen las máquinas carecen de intención, pues están basadas en su forma sintáctica no en su significado. Según lo anterior, los procesos realizados por una máquina no están dirigidos a algo, son acerca de nada (Barón, 2008).

Según Searle, los humanos hacemos relaciones con símbolos pero además podemos conocer su significado, en sus palabras “los humanos somos máquinas exponentes de programas de computación, manipulamos símbolos formalmente y aún así, podemos pensar”. Al carecer las máquinas de intencionalidad podemos decir que “no piensan”, pues la

intencionalidad deviene de la biología y claramente las máquinas no poseen tal característica. [3]

A la defensa de Turing, el científico, filosófo y académico Douglas Hofstadter contrargumenta de una manera, a mi parecer, creativa y dinámica. Esto lo hace a través de una conversación virtual (como el argumento de la habitación china) llamada "Temas metamágicos bizantinos. El test de Turing: conversación en un café" (Hofstadter, 1981).

En esta conversación virtual participan tres estudiantes: un físico, una bióloga y una filósofa. A lo largo de la conversación, lo que Hofstadter hace es evidenciar tanto sus argumentos a favor del test, como los argumentos en contra más comunes. Al igual que para Turing, para Hofstadter la percepción que en general se tiene de las máquinas es el resultado de una inducción, la cual provoca que veamos a las máquinas como herramientas inertes, poco maleables y sin vida. Para él (y para mí), si podemos romper con esta idea tajante de los alcances de una máquina, podremos entonces visualizar una inteligencia artificial capaz no sólo de pasar el test de Turing, sino de representar un nuevo paradigma en la visión científica, filosófica y psicológica de la mente humana y su relación con el entorno.

Acciones que consideramos innatas en el ser humano, como desear, pensar, intentar y esperar, están relacionadas con la intencionalidad en nuestros procesos mentales. ¿Cómo podemos entonces visualizar una máquina que pueda llegar a tener estas mismas características? Pensemos en la capacidad de información que tiene un humano, la cual es finita. No es la cantidad de información que podemos retener lo que nos hace *inteligentes*, sino cómo correlacionamos estos datos. Por lo tanto, en palabras de Hofstadter "...desear, pensar, intentar y esperar son características que emergerían de la complejización de las relaciones funcionales de las máquinas". Si bien no aclara cómo es que se va a lograr dicha complejización, lo que pone sobre la mesa es una posibilidad fundamentada en el desarrollo estructural de las máquinas que no podemos obviar, pues ha sido precisamente la aceptación de posibilidades en el pasado lo que ha permitido que con investigación y experimentación esas posibilidades sean hoy una realidad[4].

Se trata de una objeción fundamental en contra del argumento de la habitación china, pues si bien para Searle la intencionalidad emerge de nuestro sustrato biológico, para

[3] A pesar de la elocuencia del argumento de Searle, no deja claras las razones por las cuáles atribuye la intencionalidad a un proceso biológico.

[4] *Alpha Go*, es un programa de inteligencia artificial desarrollado por *Google DeepMind* para jugar el juego de mesa *Go*. Este programa ha derrotado a los mejores jugadores del mundo, sin necesidad de ninguna ventaja (*handicaps*). Observando los juegos entre *Alpha Go* y sus competidores, han calificado al programa como un jugador "conservador", con un estilo de juego diferente al que comunmente usan los humanos. El tipo de adjetivos dados a la máquina le brindan una especie de personalidad y de identificación. Esto es un ejemplo de cómo el desarrollo de la inteligencia artificial ha podido cambiar la manera de percibir a las máquinas, en unas cuántas décadas.

Hofstadter y para el filósofo de la mente Jerry Fodor (Fodor, 1997), *la intencionalidad del pensamiento existe; pero responde a la complejidad de las relaciones funcionales del ser pensante, sea este máquina o humano.*

En esta conversación los participantes se preguntan qué significaría que una máquina pasara el test de Turing, llegando a la conclusión de que lo que podríamos afirmar, es que la máquina tendría buena intuición en lo que se refiere a ser humano (afirmación que, encierra para mí un fuerte contenido de entendimiento de parte de la máquina). En otras palabras, tendríamos un eficiente simulador de pensamiento humano. Sin embargo, ¿podemos afirmar que el pensamiento simulado del ser humano es equivalente al pensamiento que se simula? Hofstadter da un ejemplo de por qué podemos afirmar algo parecido a lo anterior. Se trata del segundo argumento de Hofstadter en defensa del test de Turing.

El ejemplo toma como tema un huracán y su simulación por computadora. Si bien el huracán simulado no provoca fuertes vientos que destrozan la memoria de la máquina o la inunda físicamente, lo que podemos observar es el rompimiento de algunos lazos abstractos que han cambiado radicalmente algunos de los valores de las variables y muchas cosas más (Hofstadter, 1981). Estos fenómenos podrían ser bautizados como inundaciones, devastaciones, etc., fenómenos que en un huracán real son vistos como sus *efectos*. Entonces, ¿qué es realmente lo que se está simulando? Podríamos empezar por preguntarnos: ¿qué es un huracán y cómo lo reconocemos como tal?

La definición de "huracán" no es tan precisa como pudiéramos creer, es decir, depende de los resultados de un monitoreo de los componentes de la tormenta en cuestión y su relación con ciertos parámetros establecidos, lo que nos dirá si lo que se está viendo cumple total o parcialmente con las características de lo que se decidió en algún momento que significaba *ser huracán*. Por lo tanto, aunque no se está afirmando que el huracán simulado sea idéntico a un huracán real, lo que Hofstadter pone en claro es que hay una caracterización del huracán por sus efectos, que no depende del medio en el que el fenómeno suceda, ni de la perspectiva con la que se le mire. Estos rasgos no cambian y no dependen de los componentes materiales del fenómeno, sino de cómo se relacionan. Así, cuando analizamos las frases "huracán simulado" y "huracán *real*", nos damos cuenta de que lo único que cambia es el adjetivo que le ponemos, el cual representa el medio en el que sucede el fenómeno. Por otro lado, la palabra "huracán" es estática, esto es, un tipo de identificación del fenómeno que se genera a partir de sus efectos.

Así, lo que estamos simulando es el huracán real por medio de un huracán informático, los cuales cumplen con las condiciones para ser llamados *huracanes*, pues es una misma caracterización lo que subyace a ambos: algo que para mí, sin simulaciones, sólo *es*. En este nivel es en el que un huracán simulado tiene derecho ser llamado *huracán*, al igual

que un huracán *real*. [5] Extrapolando esta idea tenemos que *la esencia del pensamiento habría de ser el proceso que lo subyace: el cómputo matemático* (Barón 2008).

A pesar de la fuerza del argumento de Hofstadter, Searle da un contrargumento bastante convincente que exponemos a continuación.

Para John Searle, lo simulado y su simulación son axiomáticamente ajenos: "...nadie podría suponer que podemos producir leche y azúcar mediante un simulacro de computadora de la serie de procesos registrados en la lactación y la fotosíntesis". Por lo tanto, pide imaginar un programa de computadora que simula una vaca y la simulación consiste sólo en la representación de la vaca. Es claro que aunque quisiéramos, no podríamos obtener leche si *ordeñáramos* a la vaca; a lo más, podríamos obtener una representación de dicha sustancia, pero no podríamos tocarla ni beberla por más que lo intentáramos.

Searle da un ejemplo que contrapone con la simulación de un huracán, que claramente refuta la afirmación de que todo lo simulado es igual a su simulación. Afortunadamente nunca afirmamos tal cosa.

Como respuesta al argumento anterior de Searle, Hofstadter propone la siguiente situación (Hofstadter 1981): Supongamos ahora que creamos un programa de computadora que simule a un matemático y que dicho programa funcionara correctamente, ¿podríamos decir que lo que estábamos esperando recibir eran *pruebas* pero que lo único que recibimos es una representación de dichas pruebas?

La representación de una prueba constituye, en este caso, una prueba, ¿cierto? Claro, esto depende de la calidad de las pruebas representadas. Si el simulacro tuviera por objetivo producir representaciones de pruebas como las que daría un buen matematico, resultaría un colega tan valioso -en el aspecto de la representación de pruebas- como el matemático. Esta es la diferencia, según Hofstadter, entre los productos abstractos y formales como las pruebas o las canciones, y los productos concretos y materiales como la leche. ¿En qué lado de esta línea divisoria cae la mente?, ¿es la mente como la leche, o como una canción? (Hofstadter, 1981).

Si concebimos el producto de la mente como algo semejante al producto de la interacción y correlación de información, es decir, semejante a *control de cuerpo*, se diría que dicho producto es enteramente abstracto. Si lo concebimos como una especie de substancia especial, cabría pensar que su producto es concreto (Hofstadter, 1981).

Para poder responder a la pregunta anterior conviene analizar y definir puntualmente la línea que divide a los objetos abstractos de los objetos concretos. Esto, puesto que

[5] Aclaremos que no se está afirmando la igualdad entre estos dos fenómenos y mucho menos, el hecho de que en todos los casos y niveles, lo simulado pueda ser equivalente a su simulación

dicha línea tendría que asegurarnos un lugar para la simulación de "cualquier" objeto o fenómeno concreto. Sin embargo, me inclino a pensar que la mente es un material abstracto, pues si la analizamos desde la perspectiva neurocientífica, hay cosas interesantes que apuntar.[6]

El texto concluye con la siguiente observación: "todo simulacro se realiza concretamente dentro de una máquina y por medio de ésta se pretende que las representaciones de uno u otro fenómeno tengan ciertos efectos en el mundo. Si los efectos provocados por la representación del fenómeno son aproximadamente iguales a los efectos provocados por el fenómeno en sí, insistir en que se trata sólo de una representación empieza a sonar como una posición obstinada, ¿no creen?" (Hofstadter, 1981).

Más allá de mis convicciones y a pesar de que la discusión entre Searle y aquellos que intentan refutar su argumento de la habitación china continúa,[7] a la fecha no ha sido planteado un argumento lo suficientemente fuerte y consistente para mermar el debate. A lo largo de éste, tanto la filosofía como la psicología han sido campos importantes para la investigación del pensamiento de sistemas artificiales y de la función de la mente humana. Stuart Watt, investigador del Departamento de Psicología de la *Open University Milton Keynes*, siguiendo la idea de Premarck y Woodruff (Premarck & Woodruff, 1978), plantea una versión "invertida" del test de Turing, en la cual se evidencía que el papel del interrogador es mucho más importante de lo que se había pensado: "veámoslo con simpleza, una máquina aprobará el test de Turing si el interrogador *cree* que ésta posee una mente" (Watt, 1996).

Esta nueva perspectiva nos permite extender los límites del test de Turing mediante el estudio de la forma en que los humanos adjudicamos pensamiento inteligente a terceros, empezando por nuestra propia raza.

En el siguiente capítulo analizaré dicha perspectiva con dos objetivos[8]: el primero, es reforzar los argumentos a favor del test de Turing; el segundo, es dar un preámbulo de la versión del test en que tanto el imitador como el interrogador son máquinas de Turing[9], la cual, está planteada a partir de la versión del test "invertido" de Watt.

[6] *Cfr.* Liu, Xu, *et al.*, (2012, Marzo 22). "Optogenetic stimulation of a hippocampal engram activates fear memory recall". Consultado en http://www.nature.com/nature/journal/v484/n7394/full/nature11028.html

[7] *Cfr.* Searle, John. (1990), "Is the Brain a Digital Computer?", Proceedings and Addresses of the American Philosophical Association, 64 (November): 21–37 & Cole, David (Fall 2004), "The Chinese Room Argument", in Zalta, Edward N., The Stanford Encyclopedia of Philosophy.

[8] Uno de los principales propósitos de Turing al plantear su test, fue acabar con la discusión psicológica y filosófica sobre cómo funciona la mente humana. Sin embargo, me parece prudente hacer este análisis con el fin de proyectar lo acertado que es el test de Turing, incluso cuando es examinado desde diferentes planos

[9] *Cfr.* Ver Apéndice A.1.

Capítulo 2

El origen de una nueva perspectiva.

Tanto Hofstadter como Turing opinan que la existencia de pensamiento en los demás es sólo probable, y que la única forma de acercamiento es mediante su conducta, es decir, por medio de la observación de los hechos externos, como las acciones. Como ya lo hemos dicho antes, en caso de no confiar en el método de observación externa, estaríamos cayendo en un rotundo solipsismo. Pensemos en cómo es que racionalizamos el que el prójimo *esté pensando*: nos basamos plenamente en el monitoreo continuo de sus acciones, de sus procesos exteriores. Visto así, lo que hace un interrogador que participa en el test de Turing es bastante parecido.[1]

Para la Real Academia Española, "pensar.[es], entre otros, el acto de *formar o combinar ideas o juicios en la mente* e ïnteligencia.[es], entre otros, la *capacidad de entender o comprender*. Aunque pareciera loable que la respuesta a la pregunta principal de esta tesis fuera más fácil de encontrar si tuviéramos definido algún tipo de *teoría de la inteligencia*, las definiciones encontradas en la RAE nos muestran lo ambiguo que eso puede ser. Turing estaba consciente de esto y por ello propone un test que intenta filtrar sólo aquello que empíricamente buscamos para poder atribuir pensamiento inteligente a un tercero.

Como mencioné, en esta búsqueda nos basamos fundamentalmente en la conducta del observado. Sin embargo, en este reconocimiento no sólo es importante esto último, sino el tipo de conducta que el observador está esperando para poder atribuir inteligencia.

[1] La particularidad del test de Turing es que en éste no se permite ningún tipo de interacción física, lo cual, en el acto ordinario de adjudicar pensamiento inteligente a externos, es un mecanismo que usamos inconscientemente y que en general está relacionado con el antropomorfismo (Eddy et al., 1993)

En otras palabras, el patrón buscado por el observador está ligado profundamente al contexto de éste.

En primer lugar, debemos hacer la observación de que hasta donde sabemos, el grado de inteligencia más complejo que conocemos y con el cual hemos tenido contacto, es aquel que caracteriza a los humanos, o más aún, sólo hemos sido capaces de reconocer este tipo de complejidad. A partir de esto, parece lógico llegar a la conclusión de que quizás la presencia o ausencia de inteligencia no es un atributo intrínseco de todo aquello que sea capaz de pensar, sino el resultado de la interpretación que los observadores hacen al analizar el comportamiento del ser en cuestión con su entorno, es decir, que la ïnteligencia en general"no existe, pues sólo aceptaremos como ser inteligente a aquello que podamos reconocer como tal. Con éstos y otros argumentos, French (French, 1990) afirma que el test de Turing no es un test para la inteligencia, sino para la inteligencia humana orientada culturalmente[2]. Esto nos permite ver que el papel del interrogador en el test de Turing es más importante de lo que se pensaba. Su capacidad de reconocer pensamiento en externos está acotada por factores culturales y como veremos, quizás también biológicos.

El estudio de la relación de estos factores ha dado paso a una teoría psicológica conocida como "Teoría de la Mente" o como *naive psychology* (Clark, 1987), (Hayes, 1979)[3].

Esta teoría se refiere a la habilidad de reconocer estados mentales ajenos y propios, es decir, la capacidad que tenemos para ser conscientes de que el otro también es consciente, de que sabe cosas distintas a las que sabemos nosotros y de que tiene perspectivas distintas a las nuestras.[4]

Esta teoría nos interesa en cuanto a su relación con el test de Turing. Si existe dicha facultad natural para poder reconocer estados mentales en el prójimo, ésta está fuertemente relacionada con el test de Turing y dicha relación la podemos ver desde dos puntos; por un lado necesitamos que el test de Turing sea lo suficientemente fuerte para poder reconocer una Teoría de la Mente en la máquina (pues queremos intentar asegurarnos de que la máquina tiene consciencia suficiente para saber de lo que está hablando); y

[2]Esto apoya la idea de Turing que planteamos anteriormente: si la máquina piensa igual o no a los humanos, realmente no importa, pues si con su propio pensamiento logra imitar a un humano lo suficientemente bien para que lo reconozca como ser pensante, es porque en términos humanos *piensa*.

[3]Comúnmente, la definición de *Naive Psychology* (Psicología de la ingenuidad) es distinta de la de "Teoría de la Mente", sin embargo, Clark (Clark 1987) las define exactamente igual. Por lo tanto, a partir de aquí me referiré a ambas como Teoría de la Mente.

[4]*Cfr.* Baron-Cohen, S., Leslie, A. M., & Frith, U. (1985) "Does the autistic child have a 'theory of mind'?" *Cognition 21*:37-46, Premack, D. & Woodruff, G. (1978), "Does the Chimpanzee Have a Theory of Mind?", *Behavioral and Brain Sciences 1*, pp. 516–526 y Wimmer, H. & Perner, J. (1983) "Beliefs about beliefs: representation and constraining function of wrong beliefs in young children's understanding of deception." *Cognition 13*:103-128.

por otro lado, la necesidad de monitorear y controlar las tendencias que puedan existir en el interrogador debidas a su Teoría de la Mente (Caporael, 1986; Collins, 1990).

Es claro que si se pudiera reconocer que la máquina posee una teoría de mente, sería necedad de nuestra parte seguir afirmando que la máquina no puede pensar. Sin embargo, ¿es necesario que la máquina posea una Teoría de la Mente para pasar el test de Turing?

Por un lado, tenemos que es lógicamente posible para un máquina pasar el test de Turing sin poseer dicha teoría. Esto es una variación a la objeción de que no sólo no es necesario tener Teoría de la Mente para pasar el test, sino que no es necesario tener ninguna habilidad que se pueda considerar inteligente, pues sólo tiene que aparentar serlo (Block 1981).

Por otro lado, retomando el ejemplo que Turing da a la objeción de la consciencia en el que se habla con el imitador de cierto poema (página 3), Haugeland (Haugeland, 1985) afirma que la comprensión de dicho poema conlleva el entendimiento de otras mentes y por lo tanto la apropiación de una Teoría de la Mente.

Así, a pesar de que no podemos afirmar rotundamente que la máquina necesite tener Teoría de la Mente para pasar el test de Turing, según el argumento de éste, podemos decir que al menos necesita aparentar tener una.

De esta manera es que, basándose en los trabajos de Premarck y Woodruff (Premarck & Woodruff, 1978), Watt afirma que el papel del interrogador es igual, o quizá más importante que el del imitador. Para exhibir parcialmente esto, da los siguientes ejemplos.

Recordando que Turing establece el uso de una pantalla para la comunicación entre el interrogador y los agentes observados, analicemos cómo es que incluso el uso de una pantalla puede ser un factor que predisponga al observador. Hofstadter (Hofstadter, 1985) formula una versión en la que un estudiante es el imitador, su papel es simular a una inteligencia artificial pero siendo él mismo, es decir, sin pretender simular cuestiones mecánicas. A pesar de esto, ninguno de los agentes observadores en el experimento sospecharon que estaban interactuando con un humano y no con una máquina. Lo importante en esta versión es que los observadores eran personas que estaban profundamente familiarizadas con la inteligencia artificial. Esto es claramente un factor importante para su decisión, puesto que están predispuestos a la idea de que es totalmente plausible comunicarse (similarmente a como se haría con un humano) con una computadora por medio de una pantalla. Quizás el resultado sería diferente si esta vez se consideraran personas que a lo más interactúan con una computadora para checar su correo electrónico (Watt, 1996).

Otro ejemplo interesante es la interacción con el programa PARRY, creado por el psiquiatra y experto en inteligencia artificial, Kenneth Colby (Colby, 1981). PARRY es un programa que simula a un paranóico que cree que la mafia lo persigue. Colby limita a PARRY a responder preguntas que tienen sentido sólo en el caso de un interrogatorio a una persona paranóica, y reúne a un equipo de psiquiátras para que interactúen con PARRY y con paranóicos reales por medio de un teletipo sin que estos sepan que entre los pacientes hay un programa. Después de las interacciones, Colby entrega por escrito los interrogatorios a expertos en psquiatría, esta vez diciendo que uno de los escritos pertenece al interrogatorio con una máquina, desafiándolos a decir cuál de éstos es. Al no poder diferenciar entre los interrogatorios y argumentando que no fue justo que no se les avisara con anticipación la participación de una máquina, Colby repite el experimento, esta vez con la forma del test de Turing, es decir, el psquiátra en turno separado de PARRY y del paranóico real, y con la premisa de que uno de los dos es una máquina. El resultado no fue mucho mejor que el anterior. Ahora bien, uno de los psiquiatras se tomó la libertad de hacer preguntas que no fueran del tipo convenido, las cuales son éticamente aprobadas para hacerlas a un paciente que sufre de paranoia (puesto que no se quiere de ninguna manera alterar al paciente). Las respuestas que PARRY dio a este tipo de preguntas fueron totalmente sin sentido, sin relación alguna con el tema en cuestión pero sí con el asunto de la mafia. Por supuesto que es probable que este tipo de respuestas fueran generadas por un paranóico real. Sin embargo, sabiendo ahora que hay un impostor entre los interrogados, las respuestas de PARRY lo hicieron sospechoso a los ojos de los psiquiátras. (Watt, 1996)

De nuevo tenemos un paradigma en donde el contexto del interrogador afecta sobremanera sus decisiones. Es precisamente su Teoría de la Mente, fundamental en el desarrollo del cuestionamiento, pues esta capacidad (aunque probablemente intrínseca), se altera con las experiencias que se tienen a lo largo de la vida.

El estado actual de la cuestión (Eddy et al., 1993) nos dice que los únicos seres vivos que conocemos, que generan y apropian una Teoría de la Mente, son los seres humanos. Así, si se quiere analizar el potencial de una máquina, parecería lógico ponerlo ahora en el papel del interrogador. Esto es lo que hace Watt con su versión del test de Turing, en donde pone a prueba a la máquina pero esta vez en el papel del interrogador.

Lo que Watt propone es un test de Turing invertido, que intentará sanar el problema de la predisposición del interrogador debido a su Teoría de la Mente. Esta versión se basa en una prueba que consistirá en checar si un sistema atribuye estados mentales a terceros de la misma manera en que lo hacen los humanos. En palabras de Watt, “a system passes if it is itself unable to distinguish between two humans, or between a human and a machine that can pass the normal Turing test, but which can discriminate

between a human and a machine that can be told apart by a normal Turing test with a human observer".[5]

Para Watt, esta versión es una versión mejorada del test de Turing, en la cual se arreglan problemas como las posibles predisposiciones que pueda tener el interrogador, debido a su Teoría de la Mente.

Sin embargo, si analizamos la estructura que Watt propone para su test, nos damos cuenta de que podríamos adaptarla a la estructura original del test de Turing. En el interrogatorio se le puede pedir a la máquina que diferencíe a través de sus cuestionamientos con un humano y con una máquina. De hecho, se le podría pedir a la máquina que haga el papel del interrogador en un test en donde tendrá que decidir si está interactuando con una máquina o con un humano; todo esto como parte de una pregunta o petición dentro del test original.

Más aún, podríamos convertir al test en una prueba que tenga dos direcciones: una en la que el humano juzga a la máquina y otra en la que la máquina juzga al humano. Esto se puede lograr si se programa a la máquina no sólo con el objetivo de que engañe a su interrogador, sino par que, después de cierto tiempo, tenga que decidir si su interlocutor es una máquina o un humano. Con este nuevo requisito se necesitarían nuevas reglas. Por ejemplo, parece justo que así como no se le puede pedir a la máquina que realice exposiciones físicas, tampoco se le pueda pedir al humano que realice cosas como hacer una multiplicación no trivial de cinco dígitos por cinco dígitos en un segundo, puesto que tanto esto como las exposiciones físicas no son necesariamente evidencias de pensamiento inteligente.

De esta manera, dentro del mismo interrogatorio en el test original, se puede monitorear la habilidad de la máquina para atribuir estados mentales, que es lo que Watt propone a grandes rasgos.

Por lo tanto, para mí, lo que hace Watt es detallar y hacer visible una noción y una perspectiva que no habían sido analizadas anteriormente, y que son de suma importancia para el test. En otras palabras, Watt nos hizo darnos cuenta de que el papel del interrogador es más significativo de lo que pensabamos y que necesita estudiarse a detalle desde varias perspectivas, entre ellas la psicológica, pues esto probablemente nos ayude a delimitar con precisión los alcances y las consecuencias del test de Turing.

Así, me parece importante observar que realmente lo que estamos haciendo con ambos *tests*, es recopilar información de manera inductiva que nos ayude a deducir si la máquina

[5]Un sistema pasará el test si él mismo es incapaz de distinguir entre dos humanos, o entre un humano y una máquina que pueda pasar el test de Turing original, pero, que se capaz de distinguir entre un humano y una máquina que no pueda pasar el test de Turing original con un interrogador humano.

en cuestión es o no un ser pensante, pues incluso si las máquinas no llegaran a pasar el test de Turing, el que sean consideradas inteligentes será una cuestión más cultural que técnica (Watt, 1996).

Basándose en el ya explicado test ïnvertido"de Watt, Yuzuro Sato y Takashi Ikegami proponen una versión del test en el que el interrogador es una máquina de Turing. (Sato e Ikegami, 1999 y 2004).

El éxito o fracaso de una máquina de Turing como interrogador, tiene consecuencias de interés para nuestro análisis tanto de la mente humana como de los procesos internos de ciertos sistemas. En el capítulo 3 mostraremos que existen límites en las máquinas de Turing, respecto a su capacidad de discriminación entre su propio pensamiento y el pensamiento humano. En términos psicológicos, las máquinas de Turing no son capaces de diferenciar entre los estados "mentales"de una máquina y un humano, es decir, en caso de que la tuvieran, la Teoría de la Mente de una máquina no sería diferenciable de la de los humanos, si el juez es una máquina de Turing.

Capítulo 3

Máquina contra máquina.

Dentro del nuevo escenario en que el interrogador empieza a tener un papel más relevante en la historia, parece sensato preguntarnos qué pasará si el papel del interrogador en el test de Turing lo desempeña una máquina.

Lo que podemos asegurar hasta este momento es que no conocemos la existencia de un ser humano capaz de distinguir humanos de máquinas en todos los casos, a través del test de Turing.

Si hubiera una máquina capaz de distinguir humanos de máquinas a través del test de Turing, entonces sabríamos que este proceso de distinción puede ser implementado por un algoritmo. Dicho algoritmo, con toda seguridad, sería bastante complejo, pareciendo imposible que un humano lo pudiera emular eficientemente.

Sato e Ikegami,[1] intentan demostrar la imposibilidad de la existencia de dicha máquina, afirmando de este modo que si existiera un proceso por el cual se pueda diferenciar humanos de máquinas por medio del test de Turing, éste no sería recursivo.

El problema con estos autores, es que las demostraciones que ofrecen (publicadas en 1999 y 2004), presentan varios errores que expondremos a continuación, para después dar una correcta demostración de este hecho.

El teorema a demostrar es el siguiente:

Teorema 3.1. *No existe una máquina de Turing perfecta*[2] *que juegue el papel del interrogador en el test de Turing.*

[1] Sato, Yusuru & Ikegami, Takashi, 2004 & 1999, "Undecidability in the Imitation Game", Minds and Machines, 14: 133-143.

[2] *Cfr. Ver* Apéndice A, Definición A.8.

La diferencia principal entre los dos artículos de Sato e Ikegami, es que en el primero (1999) se quiere demostrar que no existe una máquina de Turing perfecta que pueda diferenciar entre cualquier máquina de Turing y cualquier humano. Por otro lado, en el segundo artículo (2004), lo que quieren demostrar es la inexistencia de una máquina que pueda diferenciar entre una máquina de Turing específica y cualquier humano. La diferencia radica en que en la segunda versión se le da al interrogador un tipo de "ventaja", al brindarle al inicio del juego el índice de la máquina de Turing con la que probablemente se enfrenta.

A continuación, se analizan ambos artículos con el objetivo de exponer los errores e inconsistencias que encontré en ellos.[3]

Las premisas que usan para sus demostraciones son las siguientes:

1. *El imitador (A) es una máquina de Turing.*
2. *El interrogador (C) es una máquina de Turing.*
3. *El jugador (X) es o un imitador (A) o un humano (B).*
4. *Al interrogador (C) se le da el índice del imitador (A).*[4]
5. *Al humano (B) no se le da el índice del imitador (A). Por lo tanto, él o ella no puede emular al imitador (A) intencionalmente.*
6. *El interrogador (C) se detiene y devuelve* `SI` *cuando ha llegado a la conclusión de que el jugador (X) es la máquina objetivo* M[5]*, y cuando la conclusión es que se trata de un humano, devuelve* `NO`*.*
7. *Si el imitador (A) no se detiene, el interrogador (C) no tiene que detenerse.*
8. *El imitador (A), el interrogador (C), el humano (B) y el teletipo usado para la comunicación, usan el mismo alfabeto de entrada y salida* Σ*.*
9. *Podemos agregar un carácter delimitador "*$\diamond$*" al alfabeto* Σ*, sin perder generalidad.*

Ejemplos sobre la estructura del juego de la imitación en las dos principales versiones son representadas en la Figura 2.

[3]En las siguientes dos secciones, usaré letra en estilo itálico para los textos citados de los artículos de Sato e Ikegami. El estilo normal se usará para exponer con mis palabras, el análisis de cada artículo.

[4]En el primer artículo esto no se supone, pues el interrogador decidirá entre un humano y cualquier máquina de Turing, no una en particular.

[5]O en el caso del primer artículo, se detendrá cuando llegue a la conclusión de que es una máquina de Turing cualquiera.

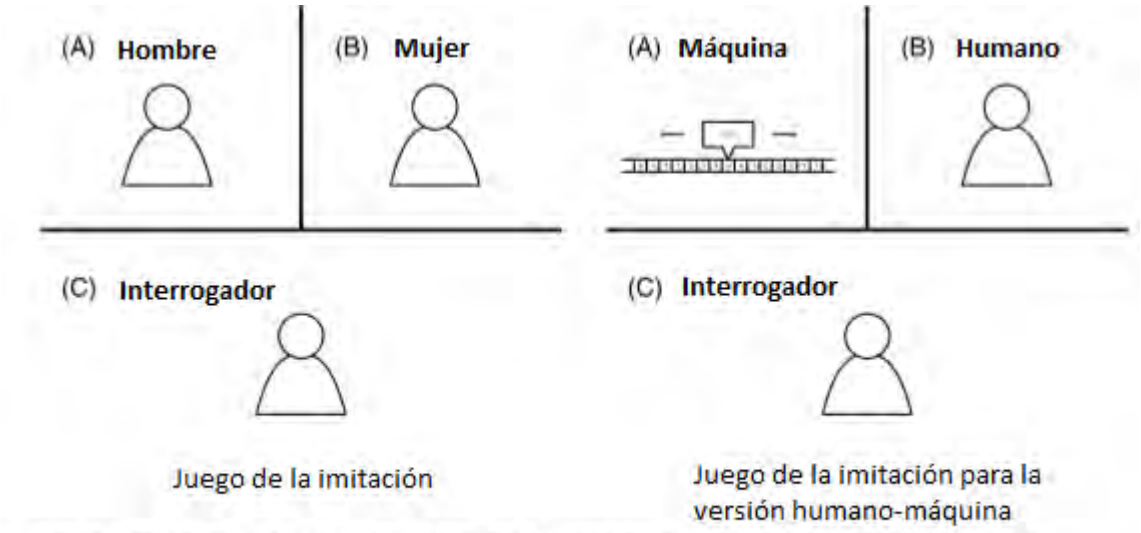

Figura 1. El juego de la imitación: un hombre (A), una mujer (B) y un interrogador (C) son los jugadores. El hombre es sustituido por una máquina en la imagen derecha.

Asumimos que el imitador intentará responder $a_i \in \Sigma^*$ *a la* i*-ésima pregunta del interrogador* $q_i \in \Sigma^*$*. El diálogo entre el imitador y el interrogador es caracterizado por medio del flujo de las sucesiones* q_i *y* a_i *como en la Figura 2. Es decir tienen la forma* $d = q_i \diamond a_i \diamond q_2 \diamond a_2 \diamond ... \diamond q_i \diamond a_i$.

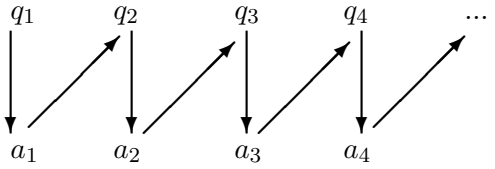

Figura 2: Flujo de los diálogos

Lo que Sato e Ikegami intentan, es modelar el test de Turing a través de un lenguaje y demostrar por reducción al absurdo que dicho lenguaje es indecidible[6].

Esta técnica es bastante recurrente en teoría de la computación. Al respecto, existen varias maneras de proceder, por ejemplo, usando reducciones,[6] o simplemente llegando a una contradicción directa.

Uno de los problemas principales, es que el segundo lenguaje que proponen implica directamente el problema de la detención, que sabemos es indecidible.[7]

[6] *Cfr.* Ver Apéndice A.2.

[7] A partir de este momento y a lo largo de este capítulo, serán necesarios elementos técnicos de computabilidad que pueden ser consultados en el apéndice o, en su defecto, en los siguientes libros: a) Martin, John C. (2011) *Introduction to languages and the theory of computation Fourth Edition*, McGrawHill. b) Cutland, Nigel. (1980) *Computability: An Introduction to Recursive Theory*, Cambridge University Press.

3.1. Análisis del primer artículo

Sato e Ikegami definen sucesiones de preguntas y respuestas de la siguiente manera:

$C \subset (\Sigma + \diamond)^*$, $C = \bigcup_{i=1}^{\infty} C_i$, donde

$$C_i = \begin{cases} \Sigma^* & (i = 1) \\ \Sigma^*(\diamond\Sigma^* \diamond \Sigma^*)^i & (i > 1) \end{cases}$$

$c_i = q_1 \diamond a_1 \diamond q_2 \diamond a_2 \diamond ... \diamond a_{i-1} \diamond q_i \in C_i$

Aquí, $c_i \in C_i$ es uno de los posibles interrogatorios de longitud i.[8]

Analizaremos ahora, el lenguaje que dan en su primer artículo, junto con las premisas que asumen en esta versión.

Para demostrar el teorema, definen un problema de decisión que llaman "TEST".

Definición 3.2. *(TEST) Definimos un lenguaje L como sigue; $L = \{c' \in (\Sigma^* \diamond \Sigma^*)^i \mid i \geq 1$, Para la entrada $c_i \in C_i$, existe una máquina de Turing M y $c' = c_i \diamond M(c_i)$[9]$\}$. La respuesta $a_i = M(c_i) \in \Sigma^*$ es generada de tal manera que la máquina de Turing que juega como imitador, intente engañar al interrogador en el i-ésimo interrogatorio. Si el discurso entre el imitador y el interrogador pertenece a este conjunto L, el interrogador decidirá que el imitador es una máquina. El problema de decisión "TEST" asociado con este juego de la imitación queda declarado como "Dado $x \in (\Sigma^* \diamond \Sigma^*)^*$, ¿$x \in L$?".*

Lo que sigue es demostrar que "TEST" es indecidible.

Así, proceden a suponer que "TEST" es decidible a lo más en el n-ésimo interrogatorio[10].

En conformidad, existe una máquina de Turing C que juega en el papel del interrogador y que puede decidir para algún $n \in \mathbb{N}$ si el oponente es una máquina o un humano, en el interrogatorio n-ésimo. Dicha máquina de Turing incluye un programa como subrutina llamado `IS-MACHINE`*;*

$$\texttt{IS-MACHINE}\ (c_i \diamond a_i) = \begin{cases} \texttt{SI} & (si\ c_i \diamond a_i \in L) \\ \texttt{NO} & (otro\ caso) \end{cases}$$

[8]Para un diálogo $c_i = p_1 \diamond r_1 \diamond p_2 \diamond r_2 \diamond ... \diamond r_{i-1} \diamond p_i \in C_i$, definimos su longitud como i. Un detalle en la notación: para que puedan existir diálogos de longitud 2, la concatenación en la definición de C_i tendría que estar elevada a la $(i-1)$ potencia.

[9]Por $M(c_i)$ entendemos la cadena de salida que M devuelve al darle la entrada c_i, en caso de que se detenga

[10]Supongo que a lo que se refieren con "a lo más en el n-ésimo interrogatorio" es que como están suponiendo que C es perfecta, entonces para cada máquina de Turing M, existe un $n \in \mathbb{N}$ tal que en el n-ésimo interrogatorio a M, C se detiene para decir que el jugador (X) es una máquina de Turing.

IS-MACHINE puede resolver correctamente si dada una cadena arbitraria $c_i \diamond a_i$, ésta pertenece o no al lenguaje L y además, siempre se detiene en un número finito de pasos. Usando IS-MACHINE, el interrogador se detendrá para decir que el imitador es una máquina cuando IS-MACHINE devuelva SI. Si el interrogatorio actual es menor al n-ésimo interrogatorio e IS-MACHINE devuelve NO, el interrogador hará una nueva pregunta. Si el interrogatorio actual es el n-ésimo interrogatorio e IS-MACHINE devuelve NO, el imitador se detiene para decir que el interlocutor no es una máquina.

El error en esta parte de la prueba se debe a la equívoca manera de definir el comportamiento de *IS-MACHINE*.

Supongamos que el juego se está dando entre C el interrogador perfecto, y una máquina de Turing M. Entonces, el interrogador genera la primer pregunta $q_1 \in C_1$, que es enviada a M. Ahora, si M no se detiene para la entrada q_1, C es incapaz de decidir (por esta razón nos importan sólo las máquinas *respondonas*[11]). Por otro lado, si $M(q_1) = a_1$, en donde a_1 es la respuesta que M manda a C, entonces, por definición de IS-MACHINE, como existe una máquina de Turing, a saber M, que para la cadena $q_1 = c_1 \in C_1$, $M(q_1) = a_1$, entonces, $q_1 \diamond a_1 \in L$. Así, cuando C reciba la respuesta a_1 y use IS-MACHINE para saber si $q_1 \diamond a_1 \in L$ o no, IS-MACHINE devolverá SI, y por lo tanto, el interrogador se detendrá para decir que el jugador (X) es una máquina de Turing.

Observemos que esto pasará para toda máquina de Turing M que converja[12] respecto a la primera pregunta hecha por C. Por lo tanto, nunca podrá existir un diálogo de longitud mayor a 1, pues si éste ya existe es porque IS-MACHINE contestó SI con la entrada $q_1 \diamond a_1$ y así, el interrogador se habrá detenido para devolver que el jugador (X) es una máquina de Turing.

De este modo, no sólo estaríamos diciendo que nuestra máquina interrogadora es perfecta, sino que sólo necesita hacer una pregunta para poder decidir correctamente. Observemos también que existe la posibilidad de que haya un juego de la imitación en el que C y un humano (H) interactúen de la misma manera. En otras palabras, dada una respuesta a de un humano, es posible que exista una máquina de Turing que dé esa misma respuesta a. Como las máquinas de Turing son deterministas y al principio del juego la entrada para IS-MACHINE es la misma, (Λ, Λ)[13], entonces, la primera pregunta generada por el interrogador es la misma, a saber, q_1. Así, ante ese humano, IS-MACHINE tendrá que devolver SI, de la misma forma que lo hizo con la máquina M, y por lo tanto el interrogador C se detendrá para devolver que el jugador (X) es una máquina, lo cual es incorrecto y por lo tanto C no sería perfecta.

[11] *Cfr.* Ver Apéndice A, Definición A.7.
[12] Ver Apéndice A, Definición A.6.
[13] Λ es la palabra vacía

La prueba de Sato e Ikegami prosigue a partir de la existencia del interrogador perfecto, creando la siguiente máquina de Turing *DIAG*;

$$\texttt{DIAG}(c_i) \begin{cases} M_{k(c_i)}(c_i)\ \texttt{CONFIA EN MI} & (si\ \texttt{IS-MACHINE}(c_i \diamond M_{k(c_i)}(c_i)) = \texttt{SI}) \\ \texttt{NO SOY LA MAQUINA} & (otro\ caso) \end{cases}$$

Para esto, ordenamos a todos los $c_i \in C_i$ en orden lexicográfico y los enumeramos. El número de c_i correspondiente en este orden es denotado por $k(c_i)$. También enumeramos a todas la máquinas de Turing, de manera que la j-ésima máquina de Turing es representada como M_j. Así, hacemos una tabla en donde empatamos las respuestas de M_i a las sucesiones c_i;

	c_i^1	c_i^2	c_i^3	...	c_i^k	...
M_1	a_i^{11}	a_i^{12}	a_i^{13}	...	a_i^{1k}	...
M_2	a_i^{21}	a_i^{22}	a_i^{23}	...	a_i^{2k}	...
M_3	a_i^{31}	a_i^{32}	a_i^{33}	...	a_i^{3k}	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
M_k	a_i^{k1}	a_i^{k2}	a_i^{k3}	...	a_i^{kk}	...
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

Para cada C_i, se tendría que elaborar una tabla como la anterior.

Así, damos c_i como entrada a la máquina $M_{k(c_i)}$ y generamos la sucesión $c_i \diamond M_{k(c_i)}$. Finalmente, calculamos $\texttt{IS-MACHINE}(c_i \diamond M_{k(c_i)})$.

Si $\texttt{IS-MACHINE}(c_i \diamond M_{k(c_i)})$=SI, $\texttt{DIAG}(c_i)$ devuelve $M_{k(c_i)}(c_i)$ CONFIA EN MI. En otro caso, $\texttt{DIAG}(c_i)$ devuelve NO SOY LA MAQUINA.

Ahora, Sato e Ikegami afirman que examinando el comportamiento de $\texttt{IS}-\texttt{MACHINE}(c_i \diamond \texttt{DIAG}(c_i))$, tendremos una contradicción, puesto que $\texttt{DIAG}(c_i^k)$ será diferente de a_i^{kk} para todo $k \in \mathbb{N}$. Así, concluyen que no existe máquina de Turing que corresponda a DIAG y, por lo tanto, $\texttt{IS}-\texttt{MACHINE}(c_i \diamond \texttt{DIAG}(c_i))$ deberá devolver NO al n-ésimo interrogatorio. Esto implicará que $\texttt{IS}-\texttt{MACHINE}$ es incorrecta, pues DIAG es sólo un programa, no un humano.

Para empezar con el análisis de su razonamiento, veamos qué quieren hacer: construir un máquina de Turing DIAG que difiera de cada máquina M_j en la entrada c_i^j (en la diagonal de la matriz), para así concluir que dicha máquina no puede existir y, por lo tanto, que $\texttt{IS}-\texttt{MACHINE}$ deberá deducir que DIAG no es una máquina sino un humano, demostrando que $\texttt{IS}-\texttt{MACHINE}$ no es perfecta.

Este procedimiento tiene un par de errores. El primero es que afirman que la manera en que $\texttt{IS} - \texttt{MACHINE}$ concluirá que DIAG no es una máquina, es por medio de la inspección de las tablas anteriores. Esto limita la definición de interrogador perfecto, pues si bien suponemos que dicha máquina concluye lo que queremos que concluya, no podemos saber cómo es que llega a dicha conclusión. Sato e Ikegami imponen un comportamiento a la máquina perfecta, que como hemos mencionado anteriormente, contradice nuestra hipótesis.

Por otro lado, al construir DIAG de tal manera que difiera con toda máquina M_j en la entrada c_i^j para cada $i, j \in \mathbb{N}$, afirman que la máquina interrogadora tendrá que decidir que el imitador es un ser humano. Para que esto sucediera, en el último paso del diálogo entre el interrogador y DIAG tuvo que haber pasado que $\texttt{IS} - \texttt{MACHINE}(c_j \diamond \texttt{DIAG}(c_j)) = \texttt{NO}$ con j la longitud del diálogo, para que así el interrogador concluyera que el imitador no era una máquina sino un ser humano. Sin embargo, por la manera en que DIAG está construida, converge[14] para toda entrada c_i. Así, es imposible que $\texttt{IS} - \texttt{MACHINE}(c_j \diamond \texttt{DIAG}(c_j)) = \texttt{NO}$ con j la longitud del diálogo, pues $c_j \diamond \texttt{DIAG}(c_j) \in L$.

Ahora, si analizamos el comportamiento de DIAG, también encontramos ciertas fallas:

Cuando DIAG recibe la entrada c_i, entonces va y revisa qué devolvería $\texttt{IS} - \texttt{MACHINE}$ con la entrada $c_i \diamond M_{k(c_i)}(c_i)$.

En caso de que $\texttt{IS} - \texttt{MACHINE}(c_i \diamond M_{k(c_i)}(c_i)) = \texttt{SI}$, entonces DIAG cambia su respuesta por $M_{k(c_i)}(c_i)$ CONFIA EN MI. Sin embargo, esto no implica que $\texttt{IS} - \texttt{MACHINE}(c_i \diamond M_{k(c_i)}(c_i)$ CONFIA EN MI$)=\texttt{NO}$, pues si existe $j \in \mathbb{N}$ tal que $M_j(c_i) = M_{k(c_i)}(c_i)$ CONFIA EN MI, entonces $\texttt{IS} - \texttt{MACHINE}(c_i \diamond M_{k(c_i)}(c_i)$ CONFIA EN MI$)=\texttt{SI}$. Y así, $\texttt{IS} - \texttt{MACHINE}$ habría contestado correctamente.

Por otro lado, si $\texttt{IS} - \texttt{MACHINE}(c_i \diamond M_{k(c_i)}(c_i)) = \texttt{NO}$, entonces $\texttt{DIAG}(c_i)=$NO SOY LA MAQUINA. Sin embargo, esto no implica que $\texttt{IS} - \texttt{MACHINE}(c_i \diamond$ NO SOY LA MAQUINA$)=\texttt{NO}$, pues de la misma forma, si existe $j \in \mathbb{N}$ tal que $M_j(c_i)=$NO SOY LA MAQUINA, entonces $\texttt{IS} - \texttt{MACHINE}(c_i \diamond$ NO SOY LA MAQUINA$)=\texttt{SI}$, y de nuevo, $\texttt{IS} - \texttt{MACHINE}$ habrá contestado correctamente. De hecho, la existencia de DIAG, implica que $\texttt{IS} - \texttt{MACHINE}$ contestará SI.

Este tipo de inconsistencias se deben nuevamente, a que no podemos dar una definición constructiva de un interrogador perfecto.

Por ejemplo, el que una máquina pueda saber si con cierta respuesta x que dé al interrogador, la van a descubrir, esto no quiere decir que si cambia su respuesta por una

[14] Ver Apéndice A, Definición A.6.

distinta de x, digamos y, entonces el interrogador ya no la vaya a descubrir. Para asegurarse de esto, tendría que volver a revisar el comportamiento de $\texttt{IS} - \texttt{MACHINE}$, al dar la respuesta y.

Por las razones ya explicadas, el primer artículo de Sato e Ikegami, no prueba el teorema.

3.2. Análisis del segundo artículo

En este artículo, Sato e Ikegami intentan demostrar el mismo teorema, pero cambian la definición del lenguaje L, y el comportamiento de la máquina `DIAG`.

Esta vez, asumen todas las hipótesis dadas al principio de este capítulo y definen correctamente las sucesiones de interrogatorios:

$C \subset (\Sigma + \diamond)^*$, $C = \bigcup_{n=1}^{\infty} C_n$, donde

$$C_n = \begin{cases} \Sigma^* & (n = 1) \\ \Sigma^*(\diamond\Sigma^* \diamond \Sigma^*)^{n-1} & (n > 1) \end{cases}$$

$c_n = q_1 \diamond a_1 \diamond q_2 \diamond a_2 \diamond ... \diamond a_{n-1} \diamond q_n \in C_n$

Aquí, $c_n \in C_n$ es uno de los posibles interrogatorios de longitud n.

A continuación, definen un nuevo problema de decisión "TEST", a partir de un nuevo lenguaje;

Definición 3.3. *(TEST) Definimos un lenguaje L_M como sigue: $L_M = \{c' \in \Sigma^*(\diamond\Sigma^* \diamond \Sigma^*)^{n-1} \mid n \geq 1,$ dada una máquina de Turing fija $M,$ si M se detiene con la entrada $c_n \in C_n,$ entonces $c' = c_n \diamond M(c_n)\}$.*[15]

Así, el problema de decisión "TEST" asociado a este juego de la imitación es "Dada M y $x \in \Sigma * (\diamond\Sigma * \diamond\Sigma*)^n$, ¿$x \in L_M$?".

Ahora, suponen que (TEST) es decidible y proceden con la demostración.

Asumimos que (TEST) es decidible en el n-ésimo interrogatorio, que se tienen ternas (M, c_n, a_n) y que el jugador (X) regresa una respuesta en el n-ésimo interrogatorio. Así, existe C máquina de Turing que juega como interrogador en el juego de la imitación. Ésta puede decidir si el oponente es la máquina objetivo M o no y siempre se detiene en una cantidad finita de pasos. Dicha máquina contiene un programa como subrutina llamada `IS-MACHINE`:

[15] Para que el lenguaje L_M tenga sentido, c' debería estar en $\Sigma * (\diamond\Sigma * \diamond\Sigma*)^n$

$$\texttt{IS-MACHINE}(M, x, y) = \begin{cases} \texttt{SI} & (si\ x \diamond y \in L_M) \\ \texttt{NO} & (otro\ caso) \end{cases}$$

IS-MACHINE puede decidir correctamente si dada la entrada (M, x, y), $x \diamond y$ está en L_M o no. Si M se detiene con la entrada x y devuelve y, IS-MACHINE decide que el jugador (X) es la máquina objetivo M y devuelve SI. Si M se detiene con la entrada x y devuelve z con $(z \neq y)$, IS-MACHINE decide que el jugador (X) no es la máquina M y devuelve NO. En otro caso, si M no se detiene con la entrada x, IS-MACHINE decide que el jugador (X) no es la máquina objetivo M y devuelve NO (pues de hecho, el jugador (X) respondió y).

Al igual que en el artículo anterior, podemos ver que uno de los errores en la definición del comportamiento de IS-MACHINE es que, una vez generada la primer pregunta q_1[16], si $M(q_1) = a_1$ entonces IS-MACHINE decidirá que el jugador (X) es la máquina M. Sin embargo, si en este mismo juego, el jugador (X) fuera un humano y no la máquina M, y éste contestara a_1 a la pregunta q_1, entonces, el interrogador decidiría que el jugador (X) es la máquina objetivo M (pues las máquinas de Turing son deterministas), y habría decidido incorrectamente.

Dado tal interrogador, construimos un algoritmo D que juegue como el imitador y que tenga el siguiente comportamiento:

$$\texttt{D}(x) = \begin{cases} \texttt{LOOP} & (si\ \texttt{IS-MACHINE}\ (x,\ x,\ \texttt{NO SOY LA MAQUINA}) = \texttt{SI}) \\ \texttt{NO SOY LA MAQUINA} & (otro\ caso) \end{cases}$$

Si IS-MACHINE$(x, x,$ NO SOY LA MAQUINA$)$=SI, D cae en un ciclo infinito, de otra manera, D devuelve NO SOY LA MAQUINA al interrogador.

A partir de esto, afirman que podemos encontrar una contradicción al analizar el comportamiento de D(D), lo cual corresponde al caso en que la entrada para D sea el diálogo D.

(i) Si D(D) se detiene y devuelve NO SOY LA MAQUINA, entonces IS-MACHINE(D, D, NO SOY LA MAQUINA) =SI, por lo tanto D(D) no se detiene. Esto es una contradicción.

(ii) Si D(D) no se detiene, entonces IS-MACHINE(D, D, NO SOY LA MAQUINA) = NO, por lo tanto D(D) se detiene y devuelve NO SOY LA MAQUINA. Esto es una contradicción.

Así, IS-MACHINE no puede ser una máquina de Turing perfecta. Por lo tanto, (TEST) es indecidible.

[16]Dado que las máquinas de Turing son deterministas y, que para generar la primer pregunta el interrogador tiene como entrada algo del tipo (i, Λ), en donde el i es el índice de la máquina M que probablemente juegue como imitador, y Λ es la cadena vacía (pues aún no existe diálogo); entonces, para todo juego de la imitación determinado por i, el interrogador preguntará la misma primer pregunta q_1 tanto a la máquina M como al humano (B).

Hay que aclarar que para llegar a esta "contradicción" se tiene que cumplir que el diálogo entre el interrogador perfecto y la máquina sea justamente el diálogo D. La probabilidad de que esto suceda está ligada a la manera en que el imitador está construido. Por lo tanto, parece improbable que una contradicción de este tipo vaya a suceder, aunque sea posible.

Más aún, la conclusión a la que llegan no es una contradicción. El razonamiento que siguen es el siguiente: para el caso (i) si `D(D)` se detiene entonces `D(D)` no se detiene; y para el caso (ii) si `D(D)` no se detiene entonces `D(D)` se detiene. Esto tiene la forma $P \Rightarrow \neg P$ o $\neg P \Rightarrow P$, las cuales son equivalentes respectivamente a $\neg P \vee \neg P$ y $P \vee P$, que claramente no son contradicciones.

Por otro lado, en el caso en que la entrada para `D` sea $x \neq$ `D`, entonces tenemos un caso análogo al del artículo anterior, pues si q_1 es la primer pregunta generada por el interrogador perfecto C, entonces, lo que `D` hace al recibir la entrada q_1 es ir y checar qué pasa con `IS-MACHINE`$(q_1, q_1,$ `NO SOY LA MAQUINA`$)$. En el caso en el que `IS-MACHINE` devuelva `NO`, entonces `D` contesta `NO SOY LA MAQUINA`. Sin embargo, esto no implica que `IS-MACHINE`(`D`, q_1, `NO SOY LA MAQUINA`)=`NO`, pues `D` revisó lo que pasaría con la máquina de índice q_1, no con ella misma.

3.2.1. Conclusiones

Analizando los errores ilustrados anteriormente, es claro que las pruebas que dan Sato e Ikegami para intentar demostrar que L y L_M son indecidibles, no están correctamente organizadas y argumentadas.

Examinando el lenguaje L_M nos damos cuenta de que podemos demostrar su indecidibilidad haciendo la observación de que implica el problema de la detención, que sabemos, es indecidible. La siguiente aclaración es suficiente.

• La manera en la que definen el comportamiento de `IS-MACHINE` y el lenguaje L_M en el segundo artículo, implica la decidibilidad del problema de la detención. Para saber si dada una cadena $x = c_n \diamond a_n$, $x \in L_M$ o $x \notin L_M$, `IS-MACHINE` tiene que saber si M se detendrá con la entrada c_n y, en caso de que sí se detenga, determinar si $M(c_n) = a_n$. Así, dada una máquina de Turing N y una cadena $w \in \Sigma^*$, podemos usar `IS-MACHINE` para saber si $w \diamond x$ está o no en L_N, para todo $x \in \Sigma^*$, es decir, para saber si N se detiene con la entrada w y en caso de que sí, checar si $N(w)$ es igual a x. Esto resolvería el problema de la detención. Por otro lado, cuando definen cómo funciona `IS-MACHINE`, afirman que en caso de que el jugador (X) responda con y a la pregunta x, entonces, si la máquina objetivo M (de la cual el interrogador C tiene el índice) no se detiene

ante la pregunta x, el interrogador C decidiría que el jugador (X) no es la máquina M, pues ésta nunca se detiene ante la pregunta x. El que `IS-MACHINE` sepa de antemano que M no se va a detener con la pregunta x para cualquier $x \in \Sigma^*$ implica también, la decidibilidad del problema de la detención.

Ahora, lo más importante de esto es darnos cuenta de que ninguno de estos lenguajes, modela correctamente al test de Turing:

(i) El lenguaje L, tiene a todos los elementos de $(\Sigma^* \diamond \Sigma^*)^i, i \geq 1$ que sean posibles diálogos entre cualquier interrogador C y cualquier máquina de Turing M.

(ii) El lenguaje L_M tiene a todos los elementos de $(\Sigma^* \diamond \Sigma^*)^n, n \geq 1$ que sean posibles diálogos entre cualquier interrogador C y la máquina M.

Ninguno de estos lenguajes tiene como únicos elementos a las diálogos que nos dan información relevante acerca del jugador (X) (cuando éste juega con el interrogador C), sino a todas las posibles conversaciones. Como vimos, el que un elemento x esté en L o en L_M, no nos ayuda a deducir correctamente si el jugador (X) es una máquina o un humano.

Algo más apropiado, sería hacer lo siguiente:

Por definición de interrogador perfecto, sabemos que para cada máquina de Turing M_i con $i \in \mathbb{N}$, existe un $n \in \mathbb{N}^+$ tal que con el diálogo de longitud n que se genere entre C y M_i, el interrogador decidirá que el jugador (X) es la máquina con índice i. De igual forma, sabemos que para cada humano (H), existe un $m \in \mathbb{N}^+$, tal que con el diálogo de longitud m que se genere entre C y el humano (H), el interrogador decidirá que el jugador (X) es un humano (H). Por lo tanto, el conjunto que nos interesa sería:

$A = \{w \in (\Sigma^* \diamond \Sigma^*)^n, n \geq 1 \mid w$ es un diálogo entre C y el jugador (X), tal que C decide con w correctamente si (X) es una máquina o un humano$\}$

Esta idea la retomaremos en la sección siguiente, en donde se da una sencilla (aunque laboriosa) demostración del teorema en cuestión.

3.3. Demostración del Teorema 3.1

Recordemos el teorema a demostrar:

No existe una máquina de Turing perfecta que juegue el papel del interrogador en el test de Turing.

Demostración:

Supongamos que existe un interrogador perfecto C para el test de Turing. Busquemos una contradicción.

Con base en C construiremos una máquina de Turing M, tal que cuando M interactúe con C, ésta *decidirá* que la máquina M es un humano. Esto probará que C no es perfecta y ésta sera la contradicción.

Puntualizamos cuáles serán las hipótesis usadas para esta demostración.

1. *El imitador (M) es una máquina de Turing.*
2. *El interrogador (C) es una máquina de Turing.*
3. *El jugador (X) es o un imitador (M) o un humano (H).*
4. *Al interrogador (C) se le da el índice del imitador (M).*
5. *Al imitador (M) se le da el índice del interrogador (C).*
6. *Al humano (H) no se le da el índice del interrogador C, ni el del imitador (M). Por lo tanto, él o ella no puede emular al imitador (M) intencionalmente.*
7. *El interrogador (C) se detiene y devuelve "$X = M$" cuando concluye que el jugador (X) es la máquina objetivo M, en otro caso, devuelve "$X = H$", ambas posibilidades al cabo de un interrogatorio finito.*
8. *Si el imitador (M) no se detiene, el interrogador (C) no tiene que detenerse.*
9. *El imitador (M), el interrogador (C), el humano (H) y el teletipo usado para la comunicación, usan el mismo alfabeto de entrada y salida Σ.*
10. *Podemos agregar un carácter delimitador "$\diamond$" al alfabeto Σ, sin perder generalidad, el cual no podrá ser utilizado en las preguntas y respuestas que se envían por medio del teletipo.*

Suponemos también una enumeración para el conjunto de las máquinas de Turing, $M_1, M_2, ..., M_i, ...$, de tal manera que cada máquina se identifique con su respectivo índice $i \in \mathbb{N}$.

Entenderemos como *pregunta* o *respuesta*[17]cualquier sucesión compuesta de elementos de Σ, es decir, cualquier elemento $w \in \Sigma^*$. Así, un *diálogo* δ de longitud n (δ_n) consiste de n *preguntas* y n *respuestas* con la forma $p_1 \diamond r_1 \diamond p_2 \diamond r_2 \diamond ... \diamond p_n \diamond r_n$. Y un *interrogatorio*

I de longitud n (I_n) consiste de n *preguntas* y $n-1$ *respuestas* con la forma $I = p_1 \diamond r_1 \diamond p_2 \diamond r_2 \diamond ... \diamond p_n$.

Sabemos que al inicio del juego, a C se la da el índice de la máquina que pudiera ser (X). Por lo tanto, podemos suponer que C dispone de dos funciones recursivas ϕ y p que trabajan como subrutinas, tal que:[18]

• $p(i, \Lambda) = p_1^i$, es la primer pregunta generada por C, cuando el índice que ésta recibió fue i. Λ respresenta al diálogo *vacío*.

• $p(i, \delta) = p_{n+1}^i$, es la enésima primer pregunta generada por C, cuando el índice que ésta recibió fue i y δ tiene longitud n.

• $\phi(i, \delta) = 0$, si el interrogador *decide* que el jugador (X) es un humano (H), con el diálogo δ.

• $\phi(i, \delta) = 1$, si el interrogador *decide* que el jugador (X) es la máquina M_i, con el diálogo δ.

• $\phi(i, \delta) = 2$ si el interrogador aún no *decide* si el jugador (X) es un humano (H) o la máquina M_i, con el diálogo δ.

Así, supongamos que C tiene el siguiente comportamiento:[19]

$$C(i, \delta) = \begin{cases} "X = M" & \text{si } \phi(i, \delta) = 1 \\ "X = H" & \text{si } \phi(i, \delta) = 0 \\ p(i, \delta) & \text{si } \phi(i, \delta) = 2 \end{cases}$$

Así, tanto C como ϕ, reciben 2 argumentos; i, que es el índice de la máquina a jugar como imitador, y δ un diálogo entre C y el jugador (X).

Llamemos Π al conjunto de todas las posibles primeras preguntas generadas por C, es decir, $\Pi = \{p(i, \Lambda) \mid i \in \mathbb{N}\}$.

Como hemos mencionado, al empezar el juego de la imitación entre C y una máquina de Turing definida M_i o un humano (H), a C se le da como argumento el índice de la

[17] Notemos que como Λ, la palabra vacía, es un elemento de Σ^*, es posible que tanto el interrogador (C) como el jugador (X) envíen por el teletipo la cadena Λ, es decir, que decidan que su pregunta o su respuesta sea la cadena vacía (algo como *pasar* en algunos juegos de mesa como el dominó). Esto es distinto a que alguna de las dos máquinas no conteste. En otras palabras, no es lo mismo no contestar nada que no contestar. Esto último es un comportamiento indeseado en cualquiera de las dos máquinas, pues C se supone perfecta y para que ésta pueda llegar a una conclusión, el jugador (X) necesita responder a las preguntas de C.

[18] *Cfr.* Apéndice A.4

[19] Notemos que al dar la definición del comportamiento de C, no caemos en el error de intentar definir cómo es que C *decide* si el jugador (X) es un máquina o un humano; simplemente llamamos ϕ al algoritmo que *decidirá* esto.

máquina M_i, es decir i. Para la construcción de nuestro imitador, supondremos que al inicio del juego, también a éste se le dará el índice de la máquina a jugar, es decir, su propio índice.

Recordemos entonces lo que queremos construir: una máquina imitadora M que engañe al interrogador C haciéndole creer que es humano. Para lograr esto, M usará su propio índice y el índice de C para encontrar un diálogo δ tal que C *decida* con la entrada (i, δ) (en donde i es el índice de la máquina M) que el jugador (X) es un humano.

Entonces, la máquina M recibirá una entrada con tres argumentos (i, j, I) en donde i es el índice de la máquina M (es decir, su propio índice), j es el índice de la máquina interrogadora C e I es un interrogatorio entre C y M.

Al inicio del juego la máquina imitadora recibirá la entrada (i, j, Λ) (en donde Λ es la palabra *vacía*). Con estos datos M devolverá (mas no enviará por el teletipo) el *diálogo* que haga que C se equivoque, supongamos $\delta = p_1 \diamond r_1 \diamond p_2 \diamond r_2 \diamond ... \diamond p_n$. Así, lo que restará por hacer es que M mande la respuesta r_k cuando reciba como entrada (i, j, I_k)[20]

Ahora, observemos que como el alfabeto Σ es finito, entonces $(\Sigma \cup \{\diamond\})$ es finito, y por lo tanto, $(\Sigma \cup \{\diamond\})^*$ es numerable.

Sea $A = \{\alpha_n \mid n \in \mathbb{N}\}$ una enumeración recursiva de $(\Sigma \cup \{\diamond\})^*$, ordenada lexicográficamente.

Notemos que todo *diálogo* entre C y una máquina M o entre C y un humano (H), es un elemento de A. Como mencionamos anteriormente, si C envía una respuesta al jugador (X) y éste nunca responde no se espera que C llegue a una conclusión. Por lo tanto, podemos suponer que C llegará a una conclusión con base en un diálogo δ, siempre y cuando el jugador (X) responda a todas las preguntas que C le envíe por el teletipo. Recordemos que al tipo de máquinas que cumplen con esto las llamamos máquinas *respondonas* con respecto al interrogador C.

A continuación se construirán los algoritmos que M necesita para obtener el *diálogo* δ que haga que C se equivoque.

Definimos un algoritmo Q, que actuará como subrutina de M cuando ésta reciba una entrada (i, j, Λ). Queremos que Q identifique a los elementos de A que son diálogos válidos[21], y que los enumere recursivamente.

$Q(0, i, j) = \Lambda$

[20] Aclaremos que el *diálogo* encontrado por M se espera que sea justamente el que C tendrá con M, así, técnicamente I_n es δ_n / r_n.

$$Q(n+1,i,j) = \begin{cases} \alpha_n & \text{si } \alpha_n \text{ es un } \textit{diálogo} \text{ válido} \\ \Lambda & \text{otro caso.} \end{cases}$$

Para definir el comportamiento de Q, supongamos que ésta cuenta con dos subrutinas, Q_1 y Q_2, las cuales se encargarán, la primera de dar una enumeración recursiva B de los elementos de A que son *diálogos*, y la segunda de dar una enumeración recursiva Δ de los elementos de B que són válidos.[23]

Análogamente a como definimos a Q, podemos definir a Q_1 y Q_2 de la siguiente manera:

$Q_1(0,i,j) = \Lambda$

$$Q_1(n+1,i,j) = \begin{cases} \alpha_n & \text{si } \alpha_n \text{ es un } \textit{diálogo} \\ \Lambda & \text{otro caso} \end{cases}$$

así, Q_1 genera una enumeración recursiva $B = \{\beta_m \mid m \in \mathbb{N}\}$ a lo más numerable, que tiene como elementos a los elementos de A que son diálogos.

$Q_2(0,i,j) = \Lambda$

$$Q_2(n+1,i,j) = \begin{cases} \beta_n & \text{si } \beta_n \text{ es un } \textit{diálogo} \text{ válido} \\ \Lambda & \text{otro caso} \end{cases}$$

y Q_2 genera una enumeración recursiva $\Delta = \{\delta_k \mid k \in \mathbb{N}\}$ a lo más numerable, que tiene como elementos a los elementos de B que son válidos.

• El comportamiento de Q_1 es el siguiente: $Q_1(n+1,i,j)$ procede a revisar la cantidad de símbolos "$\diamond$" en α_n. Si la cantidad de veces que aparece el símbolo "$\diamond$" en α_n es par, Q_1 devuelve Λ; en otro caso, recorrerá α_n de izquierda a derecha, de manera que si α_n empieza con el símbolo "$\diamond$", devuelve Λ. En otro caso, Q_1 sigue recorriendo el diálogo α_n hasta encontrar el símbolo "$\diamond$", y revisa que a éste no le siga otro símbolo "$\diamond$", es decir, Q_1 revisará que en el diálogo α_n no existan cadenas que sean elementos de $(\{\diamond\}^* - (\{\diamond\},\{\Lambda\}))$; en caso de que los haya, Q_1 devuelve Λ. Por último, si Q_1 llega al último carácter x de α_n, entonces, si $x = \diamond$, Q_1 devuelve Λ; en otro caso, devuelve α_n.

[21] Que un *diálogo* sea válido, significa que es generado por p, es decir, que si α_n es de la forma: $\alpha_n = p_1 \diamond r_1 \diamond p_2 \diamond r_2 \diamond ... \diamond p_m \diamond r_m$, entonces, para alguna $i \in \mathbb{N}$, $p(i,\Lambda) = p_1, p(i, p_1 \diamond r_1) = p_2,\ ...\ , p(i, p_1 \diamond r_1 \diamond ... \diamond p_{m-1} \diamond r_{m-1}) = p_m$. Además, dicho diálogo tiene que cumplir que $\phi(i,\alpha_n) \neq 2$ y que para todo *subdiálogo*[22] α_n^j con $1 \leq j \leq m-1$, $\phi(i,\alpha_n^j) = 2$.

[22] Dado $\beta_n = p_1 \diamond r_1 \diamond ... \diamond p_m \diamond r_m$, definimos recursivamente los *subdiálogos* de β_n, como: $\beta_n^1 = p_1 \diamond r_1$ y $\beta_n^{j+1} = \beta_n^j \diamond p_{j+1} \diamond r_{j+1}$ para $j \leq m-2$.

[23] Observemos que Q actúa como un filtro de A, es decir, a través de Q_1 filtra los elementos de A que son *diálogos* y los enumera en una lista B, para despúes filtrar los elementos de B que son *diálogos* válidos y enumerarlos en una lista Δ.

• Determinemos ahora, el comportamiento de Q_2. Lo primero que Q_2 calculará es la primer pregunta generada por C cuando recibe como entrada (i, Λ).[24] Dada p_1^i, Q_2 obtiene la longitud del *diálogo* β_n por medio de la función $d = (D + 1)/2$, en donde $D =$ cantidad de símbolos $\diamond$ en β_n. Así, si β_n es de la forma, $\beta_n = p_1 \diamond r_1 \diamond ... \diamond p_d \diamond r_d$, Q_2 compara p_1^i con p_1, de manera que si $p_1^i \neq p_1$, Q_2 devuelve Λ; de lo contrario, Q_2 revisa que para cada *subdiálogo* β_n^j con $1 \leq j \leq d-1$, si $\phi(i, \beta_n^j) = 2$, entonces $p_j^i = p_j$. Si esta implicación no se cumple Q_2 devuelve Λ; de lo contrario, Q_2 calcula $\phi(i, \beta_n)$. Así, si $\phi(i, \beta_n) = 2$, devuelve Λ, sino, devuelve β_n.

De esta manera, usando Q_1 y Q_2 alternadamente, Q genera una nueva enumeración $\Delta = \{\delta_k \mid k \in \mathbb{N}\}$, en donde los elementos de Δ son los diálogos posibles entre C y la máquina M_i o entre C y un humano (H). Dicha enumeración está ordenada bajo el orden lexicográfico, pues tanto A como B tenían el mismo orden, y Q recorrió a ambas de izquierda a derecha.

Ahora, observemos que para todo $i \in \mathbb{N}$, si i es el índice que se le da a C antes de empezar el juego, entonces existe un diálogo $\delta_k \in \Delta$ de longitud $m \in \mathbb{N}^+$, tal que $\phi(i, \delta_k) = 0$, ya que de lo contrario, C *decidiría* que todo jugador (X) es la máquina M_i, lo cual sería incorrecto si el jugador (X) es un humano (H).[25]

De esta manera, tenemos que para todo $i \in \mathbb{N}$, el siguiente conjunto es no vacío:

$Z(i) = \{\delta_k \in \Delta \mid \phi(i, \delta_k) = 0$ y para todo subdiálogo $\delta_{k'}$ de $\delta_k, (i, \delta_{k'}) \in Dom(\phi)\}$.[26]

Por lo tanto, tenemos que todo $i \in \mathbb{N}$ es *admisible para minimalización* relativa a la función $\phi : \mathbb{N}^2 \to \mathbb{N}$.[27]

Así, definimos la función $\varphi : \mathbb{N} \to \mathbb{N}$, obtenida de ϕ por *minimalización*[27], tal que:

1. $Dom\ \varphi = \{i \in \mathbb{N} \mid i$ es admisible para minimalización relativa a $\phi\}$, es decir, $Dom\ \varphi = \mathbb{N}$.

2. $\varphi(i) = \ min\ Z(i)$, para todo $i \in$ Dom φ. Esto lo escribimos como $\varphi(i) = \ min\ w\ [\phi(i, w) = 0]$.

[24] Notemos que como Q_2 conoce el índice del interrogador C, basta con imitarla en la entrada (i, Λ) para obtener la primer pregunta.

[25] Veámoslo de la siguiente manera: si yo (ser humano), me ofrezco a participar legalmente en el juego con el interrogador perfecto C cada vez que a éste se le dé un índice i, entonces para cada i, debe de existir un diálogo δ, con el cual C *decide* que soy un humano (H), es decir, con el cual $\phi(i, \delta) = 0$.

[26] Sabemos que $(i, \delta_k') \in Dom\ (\phi)$, ya que por construcción, Δ es el conjunto de todos los posibles diálogos entre C y la máquina M_i o entre C y un humano (H) que son válidos, lo cual implica que para todo subdiálogo $\delta_{k'}$ de δ_k, $\phi(i, \delta_{k'}) = 2$.

[27] *Cfr.* Ver Apéndice A.3.

Ahora, como φ es recursiva y por lo tanto computable, existe $q \in \mathbb{N}$ tal que $M_q(i) \simeq \varphi(i)$. Consideremos una máquina universal de Turing W tal que para todo $i \in \mathbb{N}$, $W(q,i) \simeq M_q(i)$.

Usando a W como subrutina, M *imitará* a W en la entrada i, obteniendo $W(q,i) \simeq M_q(i) \simeq \varphi(i) = \ min\ \delta_k \in \Delta\ [\phi(i,\delta_k) = 0]$.

Por lo tanto, como sabemos que δ_k es el mínimo diálogo tal que $C(i,\delta_k) = "X = H"$, entonces, suponiendo que el diálogo δ_k tiene longitud m, δ_k es de la forma $p_1 \diamond r_1 \diamond ... \diamond p_m \diamond r_m$ y lo que resta por hacer es que dada la pregunta p_i (es decir, si M recibe la entrada (i,j,I_i), M devuelva y envíe la respuesta r_i del diálogo δ_k.

Así, tendremos una máquina que genera un diálogo con el que el interrogador decide que el jugador (X) es un humano (H), pues para cada pregunta p_i, manda la respuesta r_i de un diálogo que lleva a C a *decidir* que se trata de un humano, provocando de esta manera que el interrogador *decida* incorrectamente.

Concretemos entonces el comportamiento de M:

1. M recibe la entrada (i,j,Λ) al inicio del juego.
2. M usa a Q con los parámetros i,j, para obtener la lista Δ.
3. M imita a W en la entrada i, es decir, $M(i) \simeq W(q,i)$, para obtener $\delta_k \in \Delta$ con $\delta_k = p_1 \diamond r_1 \diamond ... \diamond p_m \diamond r_m$, tal que δ_k es el mínimo diálogo tal que $\phi(i,\delta_k) = 0$.
4. M devuelve r_m como respuesta al interrogador C cuando reciba la entrada I_m.

Así, cuando C reciba la última respuesta r_m, $\phi(i,\delta_k) = 0$ y por lo tanto, $C(i,\delta_k) = "X = H"$, lo cual es incorrecto pues M es una máquina de Turing.

Así, C se habrá equivocado y no es perfecto, lo cual contradice la hipótesis de que sí lo es. De esta manera, demostramos que no puede existir una máquina de Turing que juegue como interrogador y que sea perfecto respecto al test de Turing. □

Observaciones:

1. El diálogo δ_k obtenido por φ nos asegura que $\phi(i,\delta_k) = 0$, es decir, que dicho diálogo según C, pertence a un juego entre él y un humano (H). Sin embargo, también nos asegura que no existe *subdiálogo* $\delta_{k'}$ de δ_k, tal que $\phi(i,\delta_{k'}) = 1$, pues por construcción, sabemos que para todo *subdiálogo* δ'_k de δ_k, $\phi(i,\delta'_k) = 2$. Esto es, los *diálogos* que son elementos de Δ, son los mínimos *diálogos* (respecto al orden lexicográfico), con los cuales C toma una decisión respecto al jugador (X).

2. Si a M se le diera un índice i' tal que $i' \neq i$, es decir, un índice distinto al suyo, entonces existiría la posibilidad de que C *decidiera* que el jugador (X) es la máquina M_i y no lleguemos a ninguna contradicción. Esto, debido a que el diálogo δ_k que M generaría, sería uno que sólo nos podría asegurar que el interrogador se *equivocaría* si estuviera jugando con la máquina $M_{i'}$, pues por construcción sabemos que $\phi(i', \delta_k) = 0$. Sin embargo, no sabemos nada de $\phi(i, \delta)$, en donde δ es el diálogo entre C y M, y el cual no es necesariamente igual a δ_k, pues las preguntas que C genera con el índice i, no son necesariamente iguales a las que genera con el índice i'. Así, lo que hicimos fue generar una contradicción usando el método de *autoreferencia* para M.

Lo que hemos demostrado es que en caso de exisitir un proceso efectivo por el cual se pueda (en todos los casos) diferenciar a un humano de una máquina de Turing a través de un interrogatorio como el propuesto por Turing, este proceso no será recursivo, lo cual iría en contra de la tesis de Church-Turing. Este resultado significaría una ruptura con el paradigma matemático y computacional sobre aquello que pensamos es *computable*.

Capítulo 4

El test de Turing: *one test to rule them all.*

En la sección anterior dimos un ejemplo de los límites de las máquinas de Turing respecto a su capacidad para distinguir entre humanos y máquinas. Este tipo de limitaciones son, para mí, una evidencia de la variedad de comportamiento que puede tener una máquina. Permitiendo esto a su vez, mermar la antaña concepción de que las máquinas son entes rígidos que sólo sirven para tareas específicas.

El test de Turing brinda la posibilidad de retar a la máquina de cuantiosas maneras, y esto para mí, es su mayor virtud. El poder de dicha prueba no se basa en que sea un test perfecto, ni en que sea el único de su especie. Para mí, su fuerza subyace en su simpleza, la cual tiene la capacidad de evidenciar las habilidades intelectuales que nos conciernen respecto al comportamiento inteligente de un ser humano. En palabras del filósofo y escritor estadounidense Daniel Dennett, "no hay nada sagrado en la elección que hizo Turing del juego de la imitación que usó para su test; es simplemente un test canónico de inteligencia, en su sentido más general".

Me atrevo a asegurar que Turing estaba consciente de la variedad de tipos de evidencia de pensamiento inteligente que podemos observar en los demás. Y, junto con ello, que podríamos reunir un vasto conjunto de pruebas para cada una de éstas. Sin embargo, Turing afirmaba que su test cumplía con la propiedad de que si alguna máquina llegara a pasarlo justa y propiamente, entonces dicha máquina sería capaz también de pasar todas las pruebas reunidas en el conjunto anterior. Como ya hemos mencionado antes, el que la máquina falle la prueba de Turing no significa que no sea capaz de realizar con éxito otro tipo de pruebas de inteligencia. Sin embargo, probablemente el éxito en la primera implicaría éxito en las posteriores. Me parece importante señalar que para mí, Turing no prentendía exponer con su test, máquinas capaces de entender temas complicados como

la *teoría de cuerdas*, o que fueran capaces de apreciar una obra de teatro; sino que en el sentido más general posible, pudieran ser consideradas *pensantes*. Esto es, no queremos a un máquina culta o brillante, sino a una que con la misma información que se le da a un humano, tenga una capacidad promedio de análisis.

Turing dió varios modelos de conversaciones de las cuales podríamos deducir inteligencia en las máquinas; un ejemplo de esto lo dimos en la página 3. Sin embargo, a lo largo de los últimos 30 años, se han gestado nuevos cuestionamientos que nos permiten probar la habilidad de respuesta de las máquinas. Dos de los más ilustrativos para mí, son los siguientes (Dennett, 1985):

Terry Winograd, líder en Inteligencia Artificial, en su intento por crear una prueba eficiente para el reconocimiento de habilidad de conversación en las máquinas, pide analizar dos oraciones que difieren sólo en una palabra.

La primera es:

El comité negó al grupo un permiso para una marcha porque ellos promovían violencia.

La segunda oración es:

El comité negó al grupo un permiso para una marcha porque ellos temían violencia.

La palabra en la cual los enunciados difieren es el verbo: promover o temer. Como Winograd resalta, el pronombre *ellos* es ambiguo en ambas oraciones. Cualquiera de las dos posibles interpretaciones es correcta. Es posible imaginar un mundo en donde los comités gubernamentales permitan y aboguen por la violencia en las calles de cierto lugar, y que por alguna extraña razón usen la falta de ésta como un argumento para negar el permiso de una marcha. Pero la interpretación natural, razonable e inteligente del primer enunciado, es que es el grupo quien está promoviendo violencia, y que en la segunda oración es el comité quien teme violencia.[1]

Si suponemos ahora que el análisis de estas oraciones se introduce en la conversación con una máquina, ésta debería responder cuál es el significado más probable en cada uno de los enunciados para demostrar pensamiento inteligente. Pero el conocimiento de sintaxis, gramática o vocabulario no sería suficiente para una respuesta loable, lo que la máquina necesitaría sería un conocimiento del mundo, de política, de comités, de grupos reaccionarios y cómo suelen comportarse y más. Necesitaría saber cómo es que el mundo funciona en general. Esto es un ejemplo de cómo puede ser usado el test de Turing, para exhibir lo mejor posible, la capacidad de análisis de una máquina.

[1] Notemos que este tipo de retos resultan ambiguos puesto que el lenguaje por su naturaleza lo es.

Es bastante común que los participantes interrogadores en el test de Turing no hagan preguntas que demanden cierto tipo de análisis inusuales, es decir, el tipo de cuestionamientos suele ser bastante simple, tanto sintáctica como semánticamente. Esto se traduce en un mal entendimiento de la rigurosidad del test y en una subestimación del mismo, pues parecería que la máquina sólo está programada para responder a preguntas simples y comunes. Aquí el reto no es sólo para el programador, sino para el interrogador, pues es este último quien delimitará el nivel de abstracción que la máquina demuestre; es él quien con las preguntas adecuadas logrará poner en evidencia el alcance del test. Recordemos que el tipo de conversación que Turing propone no está acotado por ningún parámetro, todo tipo de pregunta (que no requiera muestras prácticas) es válida. Dennett (Dennett, 1985), expone otro ejemplo en donde se desafía a la máquina:

Supongamos ahora que se le dijera a la máquina lo siguiente:

Un hombre Irlandés libera a un genio de una botella y como agradecimiento éste le ofrece dos deseos. "Mi primer deseo es una pinta de Guinness", dijo el Irlandés, y cuando la pinta apareció ante él no dudo en darle grandes tragos y maravillarse al ver cómo la pinta volvía a llenarse mientras él bebía. "¿Y cuál será tu segundo deseo?", preguntó el genio. "Oh", dijo el Irlandés, "Eso es fácil. ¡Quiero otra pinta como ésta!."
– Por favor explícame esta historia y dime si encuentras algo chistoso o triste en ella.

Ahora, podemos imaginar incluso a un niño entendiendo el chiste, sin embargo, para que esto suceda, el niño debe de tener un vasto conocimiento y entendimiento sobre la cultura del ser humano. Si bien no estamos esperando que la máquina se ría o se entretenga al leer el chiste, sí esperamos que al igual que el niño, tenga la capacidad de explicar por qué es un chiste y por qué es absurdo que el hombre pidiera otra pinta. Para esto necesitaría por igual, el conocimiento y entendimiento del mundo y sus diferentes culturas y, más específicamente, que a través de este conocimiento sea capaz de diferenciar cuándo el absurdo de cierta situación provoca risa y diversión y no indignación o enojo.

Como podemos ver, el papel del interrogador es de vital importancia. Es él quien tiene la oportunidad de hacer que el test sea riguroso y complejo. De esta manera, entenderíamos el test como una prueba que sigue extendiendo los horizontes del pensamiento inteligente del humano, al igual que el de las máquinas, no por medio de una infinita discusión, sino por medio de una cuestión pragmática y contundente.

Tanto Dennett como Hofstadter[2] han comparado el test de Turing con otro tipo de pruebas de inteligencia, como el ganar un juego de ajedrez, resolver el problema Árabe-Israelí

[2]Dennett, D. C. (1985) "Can machines think?", In: How we know, ed. M. Shafto, Harper and Row.

o robar la corona de la realeza Británica. Ante estas pruebas, concluyen que el test de Turing es mucho más eficiente y revelador por las siguientes razones: el test permite una evaluación de distintas habilidades, mientras que la prueba de ajedrez sólo mide la habilidad estratégica de la máquina; el test es iterativo, es decir, a comparación de la prueba que intenta resolver el problema Árabe-Israelí, el test puede ser repetido una infinidad de veces, además, en este caso no está claro siquiera lo que significa "resolver el problema Árabe-Israelí"; la prueba consistente en robar la corona Británica, a comparación del test de Turing, podría durar años en un sólo intento, o ser absurdamente costoso, y depender en gran medida del azar.

Ahora, podemos pensar que al igual que en la prueba del robo de la corona, podría existir un sistema que pase el test de Turing a pesar de ser simple o tonto. Dennett (Dennett, 1985) afirma que esta posibilidad es tan remota que si llegara a pasar no deberíamos espantarnos, pues es la misma probabilidad que existe en principio para cualquier test. Por ejemplo, por alguna razón la comunidad científica podría ser engañada por un test que pruebe la existencia de hierro en una muestra de oro Sin embargo, nadie duda de que es una prueba confiable la mayoría de las veces. Dennett afirma también que el análisis de este tipo de eventos suele ser opacado por un movimiento que los filósofos llaman operacionalismo.

El operacionalismo es una técnica para definir la presencia de alguna propiedad, por ejemplo inteligencia, por medio de una prueba. Para dar un ejemplo de esto, Dennett define un test que llama *el test de Dennett*, para probar si una ciudad es "una gran ciudad":

Una gran ciudad es una ciudad en la cual en cualquier día se puedan realizar las siguientes cuatro actividades:

Escuchar una orquesta sinfónica, ver un cuadro de Rembrandt, ver una competencia atlética profesional y comer *quenelles de bróchet a la Nantua* en un restaurante.

El paso operacionalista consiste en definir que una ciudad será una gran ciudad por *definición*, si pasa el test de Dennett. De esta manera, supongamos que la Cámara de Comercio en *Great Falls*, Montana, quisiera que su ciudad formara parte de la lista de grandes ciudades según el test de Dennett. Lo que tendría que hacer sería relativamente simple y barato. Por ejemplo, podrían contratar a tiempo completo 10 jugadores de básquetbol, 40 músicos, un chef que sepa preparar *quenelles* y rentar un cuadro barato de Rembrandt de algún museo (Dennett, 1985).

Según Dennett, un operacionalista tonto, aceptaría que Great Falls es ahora una gran ciudad. Por el contrario, un operacionalista racional se inclinaría a aceptar dicha afirmación, pero sólo porque tendría lo que para él sería evidencia suficiente para entender que la probabilidad de que un falso positivo como el caso de la Cámara de Comercio de *Great Falls* suceda, es astronómica. Dennett desarrolla el test suponiendo que nadie sería tan tonto y pudiente a la vez como para llevar a cabo algo como lo anterior. De hecho, en el mundo actual, una ciudad en la que puedas realizar las cuatro actividades que requiere el test de Dennett, será probablemente una ciudad en la que puedas también ir al cine, ir al teatro, comprar ropa en una tienda y muchas otras cosas que hacen a un lugar, una gran ciudad.

Las actividades elegidas por Dennett son sólo una muestra representativa de todas las posibles actividades que se podrían hacer en una gran ciudad, es decir, Dennett no las eligió porque para él fuera relevante o no que al visitar una ciudad pudiera comer *quenelles*, sino porque con esta elección incrementaba la posibilidad de que exisitieran otras actividades a realizar en la ciudad. De la misma manera, afirma Dennett, es poco razonable pensar que Turing tenía algún gusto o preferencia por los juegos de salón. En ambos tests se está haciendo una apuesta poco peligrosa: la apuesta de que aquello que pase el test, pasará muchas otras pruebas del mismo tipo, la mayoría de las veces (Dennett, 1985).

La validez de la afirmación anterior está sustentada por la experiencia, no por una demostración formal. Sin embargo, creo que es justamente este tipo de "informalidad", lo que hace al test de Turing una prueba viva y acertada. No es la falta de "formalidad", una particularidad negativa del test, sino un indicio de su carácter experimental. Y a su vez, es el tipo de experimentación ofrecida por el test, lo que lo hace tan tangible y familiar. Esto, puesto que el proceso principal de este experimento está basado en un acto que realizamos día con día: entablar una conversación con un tercero, y a partir de esto (casi automáticamente), hacer un juicio sobre ello.

Si bien no podemos garantizar cuáles fueron las razones por las que Turing eligió el método de conversación para su test, creo que por las razones expuestas anteriormente podemos pensar que este proceso de elección empezó de una manera natural. La razón para creer lo anterior, es que para mí, Turing no considera que los procedimientos teóricos tengan que ser ajenos de los procedimientos prácticos, cuando el objetivo es resolver un problema. La conversación en un juego de la imitación que quizá Turing jugó o vió jugar, pasó de ser un experimento o un juego, a un método que probablemente respondiera una pregunta que nadie se había acercado a contestar concretamente.

El vínculo entre lo práctico y lo teórico se puede ver reflejado a lo largo del trabajo de Turing, específicamente, en su labor para la decodificación de la máquina *Enigma*.

El trabajo que Turing realizó en *Bletchley Park*, como sabemos, tiene gran valor en la historia. A mi parecer, dicho trabajo fue determinado por Turing, no porque no hubiera alguien en el equipo con la capacidad suficiente para entender el funcionamiento de circuitos eléctricos, combinatoria o criptología. Así como tampoco creo que haya sido por la falta de alguien que supiera construir dichos circuitos. Turing precisa el trabajo debido a su habilidad en ambas partes.

El entendimiento de la sólida relación entre estas dos fracciones del conocimiento son, para mí, parte fundamental de cualquier cometido que busque un impacto social, en cualquiera de sus rubros o manifestaciones.

De esta manera, es que justifico el origen y el valor de una prueba, que actualmente sigue siendo una herramienta útil en campos como la inteligencia artificial y las ciencias cognitivas.

Capítulo 5

Conclusiones

En este trabajo hemos expuesto un resultado que, si bien podría parecer trivial en el sentido más general, tiene implicaciones interesantes tanto en teoría de la computación como en las matemáticas.

El resultado fue el siguiente: No existe una máquina de Turing perfecta[1] que juegue el papel del interrogador en el test de Turing.

Este teorema permite cuestionarnos acerca de la posible generalidad del resultado. Esto es, dado un sistema de cómputo[2] $s \in S$, en donde S es el conjunto de los sistemas de cómputo, ¿podemos asegurar la imposibilidad de la existencia de un sistema $s' \in S$ y $s \neq s'$, tal que s' sea un interrogador *perfecto* para el test de Turing? (Sato e Ikegami, 1999). No sólo el resultado demostrado es un ejemplo de lo anterior, sino también, la actual inexistencia de un ser humano que juegue como interrogador *perfecto* en el test.

Como hemos mencionado, el teorema nos asegura que no es posible concebir la existencia de un interrogador *perfecto* que pueda diferenciar entre máquinas y humanos. Esto implica ciertas limitaciones en las máquinas de Turing, que pueden resultar naturales si tomamos en cuenta que actualmente no ha sido posible formalizar el pensamiento humano a través de una máquina de Turing.

Por otro lado, la inexistencia de dicho método recursivo nos lleva a considerar la posibilidad de que si éste llegara a existir de manera que pueda ser llevado a cabo algún día por un ser humano, esto implicaría el dominio intelectual del humano sobre las máquinas, que Turing quizo refutar. Más aún, si este proceso pudiera ser implementado algorítmicamente, esto representaría un contrajemplo para la tesis de Church-Turing.

[1] *Cfr. Ver* Apéndice A, Definición A.8.

[2] Aquí entendemos un sistema de cómputo como cualquier sistema, no necesariamente como una máquina de Turing

Sin embargo, me inclino a creer que el razonamiento es al revés, es decir, si aceptamos la tesis de Church-Turing, entonces, el resultado demostrado implicaría que no existe un método algorítmico que podamos usar para diferenciar máquinas de humanos y por lo tanto, que a cierto nivel intelectual, las máquinas de Turing no se pueden diferenciar de los humanos. Esto es, a grandes rasgos, lo que Turing defendió en su artículo de 1950.

Las implicaciones que el test de Turing ha generado desde el momento de su publicación, no sólo son vastas sino significativas.

El resultado del análisis del origen y evolución de dicho test espera evidenciar la trascendencia de los efectos causados por éste, no sólo académica sino socialmente. Esto, dado que la aceptación de las máquinas como entes que pueden *pensar*, significaría un impacto más para la capacidad de entendimiento y misticismo de los que suele depender la raza humana.

Apéndice A

A.1. Máquinas de Turing

A.1.1. Idea intuitiva

Actualmente, los humanos interactuamos con lo que podríamos llamar máquinas de Turing "materializadas", es decir, las computadoras digitales. Al igual que los programas en nuestra computadora, las máquinas de Turing tienen una unidad de control, por medio de la cual realizan cómputos con expresiones simbólicas.

A pesar de que existen distintas caracterizaciones de las máquinas de Turing, desde el punto de vista de la teoría de la computación, todas ellas representan el mismo conjunto de funciones computables. La definición que daremos, es de las más comunes.

Una máquina de Turing trabaja por medio de un cinta infinita en un sentido, dividida en celdas iguales. Estas celdas son usadas como memoria y en ellas se realizan los computos a través de una cabeza lectora/escritora que recorre la cinta.

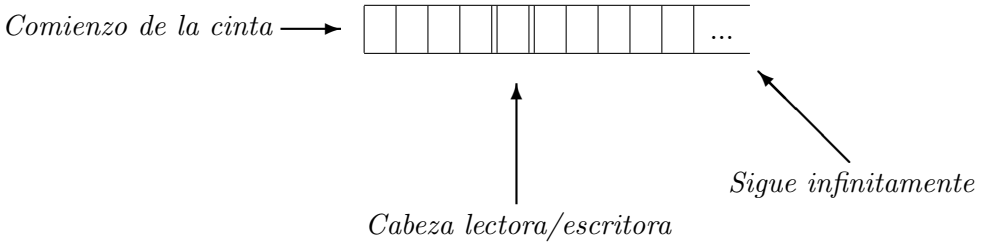

Figura 1: Diagrama de la cinta de una máquina.

La cabeza lectora/escritora inspecciona una celda a la vez, y por medio de la unidad de control (formalmente la función de transición) es dirigida a realizar ciertas acciones. Dichas acciones, son bastante sencillas:

- Leer el contenido de la celda, la cual puede estar vacía o contener un único símbolo que es tomado de un alfabeto finito predeterminado.
- Borrar el símbolo que lea en la celda.
- Escribir un símbolo en una celda vacía.
- Moverse una celda a la izquierda.
- Moverse una celda a la derecha.
- No moverse

Con estas sencillas tareas, es mucho lo que en teoría, una máquina de Turing puede realizar, siendo la capacidad infinita de memoria la diferencia principal con las computadoras digitales actuales. Asimismo, esta cualidad hace de las máquinas de Turing, objetos abstractos.

A.1.2. Descripción formal

Definición A.1. Por símbolo entenderemos cualquier forma sintáctica indivisible.

Definición A.2. Una cadena es cualquier sucesión finita de símbolos.

Definición A.3. Un alfabeto Σ es un conjunto finito no vacío de símbolos. Con Σ^* denotamos al conjunto de todas las cadenas finitas que se pueden formar con los elementos de Σ; además, en Σ incluimos a la cadena vacía ε. Llamaremos Γ al conjunto $\Sigma \cup \{B\}$.

Definición A.4. Un lenguaje $\mathcal{L}$ sobre el alfabeto Σ, es un subconjunto de Σ^*, $\mathcal{L} \subseteq \Sigma^*$.

Definición A.5. Una máquina de Turing es una sexteta $M = (\Sigma, \Gamma, Q, \delta, q_0, q_f)$ donde:

- Σ es el alfabeto de la cinta, Γ el alfabeto de la cinta y $\Sigma \subset \Gamma$.
- Q es un conjunto finito de *estados*.
- $\delta : \Gamma \times \Sigma \to \Gamma \times \Sigma \times \{I, D, \Phi\}$, es una función parcial denominada *de transición*.
- $q_0 \in Q$ es un estado especial denominado *inicial*.
- $q_f \in Q$ es un estado especial denominado *final*.

La función de transición es el *programa* de la máquina y corresponde a la unidad de control que mencionamos anteriormente. Los símbolos I, D, Φ los interpretamos como

"mover la cabeza una celda a la izquierda", "mover la cabeza una celda a la derecha" y "no mover la cabeza".

La función de transición no está definida para el estado q_f, es decir, $\delta(q_f, x)$ no está definida para ninguna x. Esto sucede porque q_f corresponde a la detención de la máquina y por lo tanto en este estado ya no es capaz de realizar ninguna operación. De la misma manera, si la cabeza lectora/escritora se encuentra al inicio de la cinta y recibe la orden de moverse una celda a la izquierda, la máquina se detiene.

Definición A.6. Sea M una máquina de Turing y sean $u, v \in \Sigma^*$. Decimos que M realiza el cómputo con entrada u y salida v, si y sólo si:[1]

- La máquina M empieza
  - Con la cadena u escrita en las primeras celdas de la cinta y las demás celdas en blanco;
  - La cabeza lectora/escritora se encuentra en la primera celda de la cinta en el estado q_0.
- La máquina M termina
  - Con v escrita en las primeras celdas de la cinta y las demás celdas en blanco;
  - La cabeza lectora/escritora se encuentra en la primera celda de la cinta en el estado q_f.

Cuando M realice un computo como hemos descrito, escribimos $M(u) = v$. Si M no realiza ningún computo con la entrada u, entonces $M(u)$ no está definido. Una segunda notación es la siguiente: si M computa v con la entrada u escribimos $M(u) \downarrow$ y decimos que M converge en u. En caso contrario escribimos $M(u) \uparrow$ y decimos que M diverge en u.

Si la máquina de Turing realiza el cómputo v con la entrada u después de n aplicaciones de la función de transición, decimos que el cómputo se realiza en n pasos o que consta de n instancias de operación.

Definición A.7. Una máquina de Turing M es *respondona* respecto a un interrogador C (máquina de Turing) para el juego de la imitación, si M responde a todas las preguntas que C le haga. Esto es, si para toda pregunta $q_i \in \Sigma^*$ que el interrogador genere y envíe por el teletipo a M, entonces, $M(q_i) \downarrow$ y si $M(q_i) = a_i$, entonces devuelve a_i como respuesta a C.

[1]Para profundizar en detalles técnicos consultar: a) Martin, John C. (2011) *Introduction to languages and the theory of computation Fourth Edition*, McGrawHill. b) Cutland, Nigel. (1980) *Computability: An Introduction to Recursive Theory*, Cambridge University Press.

Definición A.8. Un interrogador C es perfecto respecto al test de Turing, si cada vez que interroga a una máquina respondona o a un humano (que siempre responda), llega a una conclusión correcta en una cantidad finita de preguntas.

A.2. Decidibilidad

La decidibilidad de lenguajes en teoría de la computación es desde hace un tiempo, un problema importante que resolver. La idea general de un problema de decisión, es que dado un lenguaje $\mathcal{L}$ sobre un alfabeto Σ y un elemento $w \in \Sigma^*$, exista o no un método para saber si w es un elemento de L o no. Turing por su parte, demostró en 1936 que no existe un método recursivo que nos ayude a resolver el problema de la detención. A este tipo de problemas o lenguajes los llamamos *indecidibles*.

Definición A.9. Un lenguaje L sobre el alfabeto Σ, es decidible sii existe una máquina de Turing M tal que al recibir como entrada un elemento $w \in \Sigma^*$, ésta devuelva siempre un elemento fijo $x \in \Sigma^*$ si $w \in L$, o en caso contrario ($w \notin L$) devuelva un elemento fijo $y \in \Sigma^*$ con $x \neq y$.

Definición A.10. Un lengueje L es indecidible si no es decidible.

Definición A.11. Dados dos lenguajes A y B, B es reducible a A si al suponer que B es decidible, entonces podemos demostrar a partir de esto que A es decidible.

A.3. Teoría de las funciones recursivas.

El material utilizado en la demostración 3.3 de este trabajo, ocupa la teoría básica de funciones recursivas.[2]

A continuación, se dan las definiciones principales usadas en la demsotración:

Definición A.12. Un elemento $(x_1, x_2, ..., x_n) \in \mathbb{N}^n$ es *admisible para minimalización* relativa a la función $\phi : \mathbb{N}^{n+1} \to \mathbb{N}$ si el siguiente conjunto es no vacío:

$\mathcal{D}(x_1, x_2, ..., x_n) = \{m \in \mathbb{N} \mid \phi(x_1, x_2, ..., x_n, m) = 0$ y para todo $0 \leq i \leq m$
$\phi(x_1, x_2, ..., x_n, i) \in \ Dom(\phi)\}$.

El motivo por el cual se pide que para todo $0 \leq i \leq m$, la función ϕ esté definida en $(x_1, x_2, ..., x_n, i)$, es que la máquina de Turing que realice este proceso algorítmico

[2]Para más detalles se pueden consultar los siguientes libros: a) Martin, John C. (2011) *Introduction to languages and the theory of computation Fourth Edition*, McGrawHill. b) Cutland, Nigel. (1980) *Computability: An Introduction to Recursive Theory*, Cambridge University Press.

para hallar el mínimo cero, tendrá que revisar cada elemento $(x_1, x_2, ..., x_n, i)$ a partir de $i = 0$, por lo tanto si existiera j tal que $0 \leq j \leq m$ y ϕ no estuviera definida en $(x_1, x_2, ..., x_n, j)$, entonces la máquina de Turing podría no detenerse, es decir, podría diverger en el proceso de cómputo.

Definición A.13. Sea $\phi(x_1, x_2, ..., x_n, k)$ una función aritmética. Decimos que la función parcial $\varphi : \mathbb{N}^n \to \mathbb{N}$ se obtiene por *minimalización relativa* a la función ϕ, si y sólo si:

i) $Dom(\phi) = \{(x_1, x_2, ..., x_n) \in \mathbb{N}^n \mid (x_1, x_2, ..., x_n) \in \mathbb{N}^n$
admisibles para minimalización relativa a la función $\phi\}$.

ii) $\phi(x_1, x_2, ..., x_n) =$ *mín* $\mathcal{D}(x_1, x_2, ..., x_n)$, para todo $(x_1, x_2, ..., x_n) \in$ *Dom* (φ).

A.4. Construcción de una nueva máquina interrogadora

Como dijimos en el capítulo 3 de este trabajo, cuando suponemos que existe una máquina perfecta para el test de Turing, estamos suponiendo que dicha máquina tiene un comportamiento particular que no podemos saber concretamente cómo la máquina lo lleva a cabo. Sin embargo, lo que sí sabemos es que dicha máquina al ser una máquina de Turing tiene un índice que la identifica, y podemos hacer uso de éste para saber lo que dicha máquina devolvería como respuesta para cada cadena de entrada que se le dé.

El uso de las funciones p y ϕ como subrutinas del interrogador perfecto C, queda justificado de la siguiente manera:

Dada la máquina interrogadora perfecta C, podemos construir una máquina C' que de igual forma, sea perfecta para el test de Turing. Para su construcción usaremos a las funciones ϕ y φ que usamos en la sección 3.3, y supondremos que cuenta con el "programa" de C como subrutina.

Así, C' generará las preguntas que hará al jugador (X) por medio de la función $p : \mathbb{N} \times (\Sigma + \{\diamond\})^* \to \Sigma^*$, de la siguiente manera:

• $p(i, \Gamma) = p_1^1$ en donde p_1^i es la pregunta generada por C cuando se le da (i, Γ) como entrada, es decir, la primera pregunta generada por C.

• $p(i, \delta_n) = p_{n+1}^i$ en donde p_{n+1}^i es la pregunta generada por C cuando se le da (i, δ_n) como entrada, es decir, la enésima primera pregunta generada por C.

Por otro lado, definimos una función en C' $\phi : \mathbb{N} \times (\Sigma + \{\diamond\})^* \to \{0, 1, 2\}$ con la cual C' "decidirá" correctamente si el jugador (X) es la máquina M_i o un humano (H) y que tiene el siguiente comportamiento:

- $\phi(i, \delta) = 0$ si C "decide" que (X) es un humano (H) con la entrada (i, δ).
- $\phi(i, \delta) = 1$ si C "decide" que (X) es la máquina M_i con la entrada (i, δ).

Así, esta nueva máquina C' cumple con ser un interrogador perfecto que usa a C como parte de su programa para jugar correctamente en el juego de la imitación.

Bibliografía

Acero, J.J. (1995). "Teorías del contenido mental". En F. Broncano (Ed.), *La mente humana* pp.(175-206). Madrid:Trotta.

Anderlini, L. (1990), "Some Notes on Church's Thesis and the Theory of Games", Theory and Decision 29, pp. 19–52.

Barón Birchenall, Leonardo Francisco. (2008), "El juego de imitación de Turing y el pensamiento humano", Avances en Psicología Latinoamericana, [S.l.], v. 26, n. 2, p. 195-210, oct. 2009. ISSN 2145-4515.

Baron-Cohen, S., Leslie, A. M., & Frith, U. (1985) "Does the autistic child have a 'theory of mind'?" Cognition 21:37-46.

Block, N. (1981) "Psychologism and behaviourism", Philosophical Review 40:5-43

Caporael, L. R. (1986) "Anthropomorphism and mechanomorphism: two faces of the human machine". Computers in Human Behavior 2(3):215-234.

Clark, A. (1987) "From folk psychology to naive psychology", Cognitive Science 11:139-154.

Colby, K. M. (1981) "Modeling a paranoid mind". Behavioural and Brain Sciences 4:515-560.

Cole, David (Fall 2004), "The Chinese Room Argument", in Zalta, Edward N., The Stanford Encyclopedia of Philosophy.

Collins, H. M. (1990) "Artificial experts: social knowledge and intelligent machines". MIT Press.

Copeland, Jack B., (2000), "The Turing Test", Minds and Machines 10, pp. 519-539.

Cutland, Nigel. (1980) *Computability: An Introduction to Recursive Theory*, Cambridge University Press.

Darwin, Charles., (1877), "A biographical sketch of an infant", Mind 2, pp. 285-294.

Dennett, D. C. (1981) *Brainstorms: Philosophical Essays on Mind and Psychology. A Bradford book*, MIT Press.

Dennett, D. C. (1985) "Can machines think?", In: How we know, ed. M. Shafto, Harper and Row

Eddy, T. J., Gallup, G. G., & Povinelli, D. J. (1993) "Attribution of cognitive states to animals: anthropomorphism in comparative perspective", Journal of Social Issues 49(1):87-101.

Fodor, J. (1997), *El olmo y el experto. El reino de la mente y su semántica*, Barcelona: Paidós.

French, R. M. (1990) "Subcognition and the limits of the Turing test", Mind 99:53-65.

Harnad, S. (1991) "Other bodies, other minds", Minds and Machines 1(1):43-54.

Harnad, S. (1992) "The Turing test is not a trick: Turing indistinguishability is a scientific criterion", SIGART Bulletin 3(4):9- 10.

Haugeland, J. (1985) *Artificial intelligence: the very idea*, MIT Press.

Hayes, P. J. (1979) "The naive physics manifesto. In: Expert systems in the microelectronic age", ed. D. Michie, Edinburgh University Press.

Hofstadter, D.R. and Dennett, D.C. (1981), *The Mind's I*, Basic Books.

Hofstadter, D. R. (1985) *Metamagical themas: questing for the essence of mind and pattern*, Basic Books.

Martin, John C. (2011) *Introduction to languages and the theory of computation Fourth Edition*, McGrawHill.

Premack, D. & Woodruff, G. (1978) "Does the chimpanzee have a theory of mind?", Behavioural and Brain Sciences 4:515-526.

Sato, Y. & Ikegami, T. (1999), "Undecidability of the Imitation Game", Proceedings of Symposium on Imitation in Animals and Artifacts, pp. 157–159.

Sato, Y. & Ikegami, T. (2004) "Undecidability in the Imitation Game", Minds and Machines, 14: 133-143.

Searle, J. (1983). "Intentionality: An essay in the philosophy of mind", New York: Cambridge University Press.

Searle, John (1990), "Is the Brain a Digital Computer?", Proceedings and Addresses of the American Philosophical Association, 64 (November): 21–37

Searle, J. R. (1980) "Minds, brains, and programs", Behavioural and Brain Sciences 3:417-424.

Sieg, W. (2008). "Church without dogma: Axioms for computability", In B. Lowe, et al. (Eds.), New computational paradigms (pp. 139–152). New York: Springer.

Turing, A. M. (1950) "Computing machinery and intelligence", Mind LIX(2236):433-460.

Watt, S. N. K., (1996), "Naive Psychology and the Inverse Turing Test", Psychology 14(7), p.1.