

---

# Reflexiones sobre Computabilidad, Teoría y Modelos

## Reflections on Computability, Theory and Models

---

**Miguel Salinas Molina**

masepistemologia@gmail.com

Universidad Nacional Mayor de San Marcos,  
Facultad de Matemáticas, Lima, Perú

RECIBIDO: 28/10/2023 - ACEPTADO: 25/11/2023 - PUBLICADO: 30/12/2023

---

### RESUMEN

El presente ensayo es una reflexión y al mismo tiempo una propuesta sobre el concepto de programa o sistema de computadora en relación a una Teoría o Modelo; sosteniendo la existencia de dos interpretaciones diferentes con respecto a uso de la palabra Modelo, como una teorización que media entre un representante y una representación, como si fueran una relación de un objeto y su imagen. Utilizamos de definición de la máquina de Turing para clarificar la necesidad de los conceptos de Teoría y Modelo para la definición de computabilidad.

**Palabras clave:** Teoría, Modelo, Computabilidad, Máquina de Turing.

### ABSTRACT

This essay is a reflection and at the same time a proposal on the concept of a computer program or system in relation to a Theory or Model; supporting the existence of two different interpretations regarding the use of the word Model, as a theorization that mediates between a representative and a representation, as if they were a relationship between an object and its image. We use the definition of the Turing machine to clarify the need for the concepts of Theory and Model for definition of computability.

**Keywords:** Theory, Model, Computability, Turing Machine.

## I. EL PROBLEMA

Empezamos definiendo una Teoría, como un conjunto de expresiones que tienen relación y estructura; así cuando expresamos conceptos relacionados que se alinean de forma que nos ayudan a explicar un tema, incluso, a veces nos permiten predecir aspectos tratados en el tema, decimos que estamos teorizando, así queda constituida una Teoría.

Sabemos que se suele definir que una Teoría, como conjunto de proposiciones, son ideas que contienen diversos elementos, tales como: conceptos que pueden ser básicos o derivados de estos mismos, objetos que son elementos que representan la realidad o son solo conceptuales, relaciones entre los conceptos con objetos y/o entre ellos mismos.

Así también, solemos utilizar el término de Modelo, como representación o realización de una Teoría, pero sabemos, en muchos casos, se usa como Modelo a una Teoría, resultando confuso el uso conceptual entre los términos Modelo y Teoría, por ser así, pretendemos elucidar el concepto de Modelo, especialmente en relación a los programas o sistemas de computadoras que solemos construir los que nos dedicamos a desarrollar sistemas computacionales.

Para lograr explicar de mejor manera el uso del término Modelo y Teoría, nos basamos en el documento de mi autoría sobre el Concepto de Computabilidad y la Teoría de Turing, que fue la Tesis de Grado para optar el Magister en Filosofía, extrayendo de esta los elementos básicos y necesarios sobre Modelo y Teoría.

## II. MODELO COMO REPRESENTADO O REPRESENTACION.

Empezamos planteando la idea sobre el uso diverso del concepto Modelo, así lo expresa el ilustre filósofo español Ferrater Mora<sup>1</sup>: “Epistemológicamente, la noción de modelo ha sido, a su vez, empleada en varios otros sentidos” (Ferrater Mora, 2004:2433).

Así también tomamos en cuenta lo mencionado por el filósofo David Calvo<sup>2</sup> en su tesis doctoral, donde interpreta a Ferrater Mora señalando cuatro usos del concepto de Modelo: “Como modo de explicación de una realidad, como forma de presentación, como sistema que sirve para comprender otro sis-

tema y como sistema real que la teoría trata de representar” (David Calvo, 2006: 35-36), y nosotros constatamos que Ferrater Mora indica: “Se ha hablado a veces (vagamente) de modelo como de un modo de explicación de la realidad, específicamente de la realidad física. Por ejemplo, se ha hablado de <modelo mecánico> equivalente al mecanicismo... Se ha hablado asimismo de modelo como de alguna forma de representación de alguna realidad o serie de realidades, de algún proceso o serie de procesos, etc. Ejemplo de un modelo puede ser un dibujo... Un modo muy común de entender ‘modelo’ es tomar como modelo un sistema que sirva para entender otro sistema, como cuando se toma el paso de un fluido por un canal como modelo de tráfico... Otro modo de entender ‘modelo es tomar como tal un sistema del cual se trate de presentar una teoría. El modelo es entonces la realidad efectiva o supuesta- que la teoría trata de explicar” (Ferrater Mora, 2004: 2433).

Al tratar sobre “... la palabra Modelo en el diccionario de *Lógica y Filosofía de la Ciencia* de Mosterín<sup>3</sup> y Torretti<sup>4</sup>, encontramos dos formas del uso de la palabra Modelo: dos acepciones muy diferentes y en cierto modo opuestas. La primera corresponde a que un Modelo es una realización de la Teoría, esta concepción se encuentra en la denominada Teoría de Modelos, donde una interpretación M de una teoría T, es un modelo de T, de forma que los enunciados en T se cumplen en M; la segunda corresponde a que un Modelo es una representación de una situación, el Modelo resulta ser la teoría que explica una realidad” (Salinas, 2011:54-55).

También sabemos que “la definición de modelo en las ciencias fácticas (física, química y otras), suele encontrarse en relación binaria entre Modelo y realidad a la que se pretende representar: No se trata de una relación donde cada elemento del Modelo corresponde a un elemento del sistema real, por el contrario, entre ambos se establece una relación compleja, de sistema a sistema, donde algunos objetos del sistema real pueden no aparecer en el Modelo y a su vez, algunas variables del Modelo pueden no tener su correlato en el sistema real, éste es el caso de Modelos que introducen entidades teóricas, no directamente observables, cuyas propiedades no pueden ser determinadas por vía empírica en el sistema real” (Salinas 2011:56).

1 José Ferrater Mora (nacimiento: Barcelona, 30 de octubre de 1912; fallece: 30 de enero de 1991) filósofo español, entre sus obras filosóficas destaca el texto *Diccionario de Filosofía*.

2 David Calvo, Doctor en filosofía de la Universidad Complutense de Madrid, tomamos en cuenta su tesis doctoral sobre *Modelos teóricos de la Física*.

3 Jesús Mosterín (nacimiento: Bilbao, 24 de septiembre de 1941; fallece: Barcelona, 4 de octubre de 2017) filósofo y matemático español. Sus trabajos con frecuencia son de la filosofía y la ciencia.

4 Roberto Carlos Torretti Edwards (nacimiento: Santiago de Chile, 16 de enero de 1930, fallece: 12 de noviembre de 2022) filósofo con trabajos diversos sobre la historia de la filosofía y la física y matemáticas.

Así seguimos: “En las matemáticas y lógica, el Modelo se relaciona a teorías (previamente desarrollada) en el sentido de contener términos y formas de inferencia pertenecientes a la Teoría. En algunos casos se denomina “Modelo matemático” de una Teoría fáctica<sup>5</sup> a la Teoría matemática, a la cual corresponde a dicha Teoría fáctica. Se tiende a identificar la Teoría fáctica con el Modelo matemático asociado. Pero un Modelo matemático, en tanto estructura puramente sintáctica, no constituye una Teoría fáctica; para convertirse en tal requiere de una interpretación (semántica) en términos de la realidad expresada” (Salinas 2011:56).

Resaltamos lo sostenido por David Calvo, por que relaciona la realidad con el Modelo y Teoría, así lo expresa: “La relación <ser modelo de> con la letra M, sería una relación binaria tal que:  $X M Y = \langle x \text{ es modelo de } y \rangle$ . Si el modelo constituye la estructura que ejemplifica la teoría, los dos componentes  $x$  e  $y$  de esta relación son la realidad y la teoría, respectivamente.” (Calvo, David 2006: 40).

En mi investigación del 2011, establezco que “la Teoría describe a la ‘realidad’, es decir la describe cómo funciona. Esta definición puede trasladarse a las dos orientaciones que suele referirse a Modelo (en las ciencias fácticas y en las ciencias formales). Así en las ciencias fácticas el Modelo explica una determinada realidad física, mientras que en las ciencias formales el Modelo refiere a una realidad que es una Teoría matemática, es una construcción teórica que no tiene ninguna correspondencia a una realidad física y por lo tanto son ‘reales’ en ella misma... Otra forma de elucidar el concepto de Modelo es revisando la relación binaria entre lo representado y su representación, en el sentido que al representado le corresponde la representación, como una imagen. La representación está en correspondencia binaria a lo representado” (Salinas 2011:57).

Con la finalidad de facilitar lo expresado, ejemplificamos lo tratado sobre el uso del concepto Modelo en los dos sentidos expuestos: primero como lo representado, siendo el Modelo un objeto a ser producido industrialmente, hacemos primero el modelo a ser aprobado, para luego producirlo en cantidades industriales; segundo, como la representación: un mapa es el Modelo que presenta las calles y avenidas de una ciudad, las líneas están dibujadas en el mapa que tal forma que resulta ser el Modelo de la ciudad y sus calles.

<sup>5</sup> Factivo en el sentido que se refiere a situaciones del mundo, a los hechos, basado en lo que ocurre en el mundo en oposición a lo imaginario.

Sostenemos que “El Modelo entendido como representado, ocurre al considerar al objeto que sirve de observación como Modelo para obtener de imagen (es el objeto a representar), sucede cuando se hace una pintura desde una “modelo” o para obtener una fotografía... El Modelo como representación ocurre en los casos en el que se formula una teorización que explica una situación real, como sucede en la Teoría del modelo atómico (es una representación de un átomo)” (Salinas, 2011:57-58)

Afirmamos que el concepto de Modelo requiere de precisión, específicamente en el campo filosófico y lingüístico, de manera que resulta válido en las ciencias formales (lógica y matemáticas) utilizar otra palabra en referencia a “Modelo” para que contribuya a la claridad expresiva. Así lo expresa Mosterín, proponiendo la palabra “realización” en vez de Modelo: “De todos modos, y para terminar, hay que reconocer que también sería coherente usar la palabra ‘realización’ en vez de ‘modelo’ para lo que se llama modelo en la teoría de modelos y reservar la palabra ‘modelo’ para la descripción teorizada de un sistema real” (Mosterín, 2008:253)

Resaltamos “que en el campo de la lógica la teoría de modelo, es el estudio de las relaciones entre las estructuras matemáticas y los lenguajes formales; en el prólogo escrito por Mosterín en el libro de María Manzano<sup>6</sup> *Teoría de Modelos* (1989), refiere a que la teoría de modelos no es una teoría semántica que ponga en relación a los lenguajes naturales con la realidad física y social, sino una teoría matemática que pone en relación unos sistemas matemáticos con otros sistemas matemáticos. Aclara que el libro versa sobre la teoría clásica de modelos. Manzano relaciona un lenguaje con el significado de sistema, refiriéndose como sigue: ‘El esquema abstracto de la Teoría de Modelos es así: Tenemos un lenguaje  $L$  y una clase de objetos  $M$  que son sistemas, y entre estos dos tipos de realidades tendemos un puente: la noción de verdad. Este planteamiento, aparentemente tan simple, proporciona una flexibilidad y alcance a la Teoría de Modelos.’” (Salinas, 2011:59-60).

Resulta necesaria para nuestra explicación la definición que proporciona Manzano, así establece que “... el concepto de modelo necesitó precisar una teoría como expresable en un lenguaje, como el conjunto de sentencias cerradas bajo la relación de deducibilidad, es decir, una teoría es un subconjunto de sentencias de un lenguaje y si una sentencia es deducida de la teoría, entonces pertenece a

<sup>6</sup> María Manzano Arjona, (nacimiento: Archidona, enero 1950) Doctora en Filosofía, investigaciones en teoría de modelos y sistemas.

la teoría: Sea  $L$  un lenguaje de primer orden.  $T$  es una teoría de  $L$  si  $T \in \text{SEN}(L)$  y para cada  $\varphi \in \text{SEN}(L)$  se cumple: si  $T \vdash \varphi$  entonces  $\varphi \in T$ . Luego introduce el concepto de Modelo como sistema, y a un subconjunto del Modelo la denomina Teoría, así nos dice que los elementos del subconjunto satisfacen las sentencias del lenguaje  $L$ , estas sentencias forman parte de la teoría: Sea  $K \in M$ . Llamaremos teoría de  $K$  al conjunto de todas las sentencias de  $L$  verdaderas en todos los sistemas de  $K$ . Es decir,  $\text{TEO}(K) = \{ \varphi \in \text{SEN}(L) / A \text{ sat } \varphi, \text{ para cada } A \in K \}$  (Salinas, 2011:60).

Podemos concluir con respecto a la definición proporcionada por Manzano, que dado un lenguaje  $L$ , contiene sentencias que constituyen una Teoría  $T$ , donde dichas sentencias satisfacen definiciones que son parte de un sistema, de tal modo que un Modelo es una clase del sistema que satisface las definiciones mencionadas, así de lo expuesto, resulta que la teoría de Modelos es una rama de la lógica que se ocupa de la relación entre los lenguajes formales y sus representaciones basadas en estructuras.

Relacionamos los conceptos de Manzano "... en su artículo *¿Qué es esa cosa llamada lógica?*, del 2005, que dice: El puente entre estos dos tipos de realidades es el concepto de verdad; concretamente la noción de fórmula  $\varphi$  es verdadera bajo la interpretación  $F$ ... Así mismo... en su artículo *Sobre Razonamiento Formal*, publicado por la Universidad de Castilla-La Mancha, en el 2006, define la variación ocurrida en los últimos años en el campo de la lógica, habiendo ampliado su campo de acción, saliendo del formal razonamiento matemático para ser utilizado en diversos campos... también fenómenos de gestión y transmisión de información, de toma de decisiones y de la acción, y en general en casi todos los contextos gobernados por reglas" (Salinas, 2011:61-62).

Podemos concluir, de acuerdo al seguimiento conceptual realizado a Manzano, que la lógica se extiende más allá a lo que puede ser ejecutado por una máquina y que está en relación al flujo de información, en nuestro caso, nos resulta pertinente denominar sistemas de información<sup>7</sup>: "No se agota en el cálculo que un humano o una máquina pueda efectuar ya que también le interesan las interacciones entre los agentes que participan en la conversación, el proceso de adquisición de conocimiento,

7 Sistema de información, son los sistemas que toman en cuenta el almacenamiento, transmisión, recepción y producción de información. Estos sistemas están en relación a los procesos computacionales, sistemas en estrecha relación en el uso de computadora y redes de comunicación.

la dinámica y el flujo de información" (Manzano, 2006:71).

Con respecto a la lógica y modelos expresamos: "... la definición del campo de acción de la lógica con la Teoría de Modelos, en su relación a las interacciones entre los diversos agentes que participan de la conversación y/o proceso, específicamente en la dinámica del flujo de información en relación a la denominada tecnología de la información<sup>8</sup>" (Salinas, 2011:62).

### III. COMPUTABILIDAD, AUTÓMATAS Y TEORÍA DE TURING.

Empezamos establecido lo siguiente "El concepto de computabilidad en las matemáticas suele entenderse a la obtención de un valor mediante la ejecución de un cálculo. Este es definido mediante funciones que resultan ser operaciones aritméticas. Dado el desarrollo de las matemáticas en el área de las funciones y de la teoría de los números, permitieron reflexiones en temas que constituyeron la ciencia de la computación. Así en los siglos XIX y XX el razonamiento de tipo matemático se hace filosófico" (Salinas, 2011: 64), así lo menciona el filósofo Roberto Torretti<sup>9</sup>: "En los siglos XIX y XX la matemática prolifera y florece como quizás ningún otro que hacer del espíritu. Movidos por la misma riqueza y audacia de sus invenciones... Su reflexión es lo que se llama filosofía, y así la entienden; pero la conducen como matemáticos que son, aunando libertad y rigor, fantasía ubérrima y precisión pedante, en el estilo propio de su disciplina" (Torretti, 1998: XI).

Recurrimos a las reflexiones proporcionadas por Ferrater Mora que "define el concepto de computabilidad mediante funciones del tipo recursivo, incluye términos utilizados por Tarski<sup>10</sup> en relación al concepto de teoría decidible. Afirma que un procedimiento es una secuencia de operaciones que per-

8 Según la Asociación ITAA (Information Technology Association of America). La tecnología de la información es el estudio, diseño, desarrollo, implementación, soporte o dirección de los sistemas de información computarizados, en particular de software de aplicación y hardware de computadoras.

9 Roberto Torretti, (nacimiento: Santiago de Chile, 16 de enero de 1930; fallece: 12 de noviembre de 2022), autor del libro *El Paraíso de Cantor*, documento con temas de filosofía de las matemáticas.

10 Alfred Tarski, (nacimiento: Varsovia, 14 de enero de 1902, fallece: Berkeley, California 26 de octubre de 1983). Judío, adoptó su apellido al convertirse en 1923 al catolicismo. Formó parte de la importante escuela polaca de lógica y filosofía hasta 1939, en que se estableció en Estados Unidos; la emigración le salvó de la suerte de la mayor parte de su familia, que pereció bajo la ocupación nazi de Polonia. Desde Estados Unidos, enseñaría la lógica. Sus aportes en teoría de conjuntos, lógica polivalente, niveles de lenguaje y metalenguaje y conceptos semánticos. Autor de *Introducción a la lógica y a la metodología de las ciencias deductivas* en el año 1941 y *La concepción semántica de la verdad y los fundamentos de la semántica* en 1944.

mite la demostración de un teorema, y que la teoría es decidible si sus funciones son recursivas” (Salinas, 2011:64), así tenemos “Se llama decidible a un cálculo C cuando puede forjarse un método o un procedimiento mecánico mediante el cual sea posible decidir – en una serie de operaciones finita – si una fórmula bien formada de C es o no un teorema de C... si existe en un cálculo C o una teoría formalizada T un procedimiento o método de decisión es llamado problema de decisión. Si se encuentra tal procedimiento o método, el cálculo o la teoría formalizada reciben el nombre de decidibles; si no, el de indecidible... Las anteriores definiciones no tienen carácter formal. Para una definición formal suficiente del término ‘decidible’ aplicando a una teoría formalizada T usaremos la formulación de A. Tarski (*Undecidable Theories*, 1953). Una teoría es llamada decidible si el conjunto de todas sus funciones válidas es recursivo; de lo contrario, es llamada indecidible.” (Ferrater Mora, 2004, 786).

“Precisa también el concepto de decisión en el mismo sentido que el utilizado por Hilbert<sup>11</sup> en 1900 (al plantear 23 problemas de investigación). Específicamente en el enunciado del tercer problema que dice: Encontrar un procedimiento para la solución de las ecuaciones diofánticas<sup>12</sup>. El programa sugerido por Hilbert trata sobre la búsqueda de la axiomatización de las matemáticas, este sería conocido como *Entscheidungsproblem*, que significa: Problema de Decisión” (Salinas, 2011: 65).

Hilbert menciona en 1990 lo siguiente: “La compatibilidad de los axiomas aritméticos. Cuando estamos inmersos en la investigación de los fundamentos de una ciencia, debemos establecer un sistema de axiomas que contiene una descripción exacta y completa de las relaciones que subsisten entre las ideas elementales de esta ciencia. Los axiomas para configurar son al mismo tiempo, las definiciones de las ideas elementales, y ninguna declaración en el ámbito de la ciencia cuya fundamentación nos están poniendo a prueba se considera correcto a menos que pueda derivarse de esos axiomas por medio de un número finito de pasos lógicos... quiero designar los siguiente como los más importantes de las numerosas preguntas que se le puede pedir en lo que respecta a los axiomas: probar que no son contradictorios, es decir, que un número defini-

do de pasos lógicos basados en los mismos nunca pueden conducir a resultados contradictorios.” (Hilbert, 1900:31-32).

“El concepto de computabilidad en relación al de recursividad se formuló antes del planteamiento de Hilbert, y fue madurando hasta lograr la definición de una teoría, que se llamaría teoría de la recursión, contribuyendo en los años 1960 en el nacimiento de la ciencia de la computación (en Estados Unidos) o informática (en gran parte de Europa). (Salinas, 2011: 65)

Así tenemos que el concepto de “algoritmo en la teoría de Turing<sup>13</sup> está expresado implícitamente, cuando define el concepto de máquina, en el sentido de un dispositivo que ejecuta instrucciones muy básicas. Así lo dice Wittgenstein<sup>14</sup>: “Si el cálculo nos aparece como una actividad maquina, entonces la máquina es el ser humano que realiza el cálculo” (Wittgenstein, 1978:195), esta expresión resulta equivalente a decir que las máquinas de Turing son personas que calculan.” (Salinas, 2011: 114)

Nos resulta necesario el concepto de computabilidad que manifiesta Turing, que está presente en su obra, lo decimos de esta forma: “se encuentra constantemente en sus diversas publicaciones, realizadas a lo largo de su vida científica, así hemos encontramos los siguientes documentos que explican suficientemente el concepto computable en la teoría de Turing<sup>15</sup>. Los documentos son los siguientes: 1) *On computable numbers, with application to the Entscheidungsproblem* (1936), 2) *Systems of logic based on ordinals* (1938), 3) *Intelligent Machinery* (1948), 4) *Computing machinery and intelligence* (1950), 5) *Can Digital Computers Think* (1951), 6) *The Chemical Basis of Morphogenesis* (1952), 7) *The Chess* (1953) y 8) *Solvable and Insoluble Problems* (1954). (Salinas, 2011: 115)

11 David Hilbert, (nacimiento: Königsberg, Prusia, 23 de enero de 1862; fallece: Gotinga, Alemania; 14 de febrero de 1943) matemático, propone en 1900 en el congreso internacional de matemáticos en París y en 1928 en el congreso internacional de Bolonia, un procedimiento algorítmico general para resolver cuestiones matemáticas.

12 Ecuación diofántica, es una ecuación algebraica con coeficientes enteros, donde la solución de las variables son números enteros. Ésta debe su nombre al matemático griego Diofanto de Alejandría (200 DC – 280 DC).

13 Alan Mathison Turing (nacimiento: Paddington, Londres, 23 de junio de 1912; fallece: Wilmslow Cheshire, 7 de junio de 1954) matemático, lógico, informático teórico, criptógrafo, filósofo y biólogo. Considerado como uno de los padres de la ciencia de la computación. Proporcionó los conceptos de algoritmo y computación: la máquina de Turing. Formuló su propia versión que hoy es ampliamente aceptada como la tesis de Church-Turing (1936).

14 Ludwig Josef Johann Wittgenstein. (nacimiento: Viena, Austria, 26 de abril de 1889; fallece: Cambridge, Reino Unido, 29 de abril de 1951) filósofo y lingüista austriaco, posteriormente nacionalizado británico. Publicó el libro *Tractatus logico-philosophicus*, influyó en los positivistas lógicos del Círculo de Viena, del que nunca se consideró miembro.

15 Los documentos corresponden a los trabajos de investigación de Alan Turing, publicados en los años: 1936 (define la máquina Turing), 1938 (su tesis doctoral en matemáticas), 1948 (informe cuando trabajaba en NPL) y 1950 (presenta el conocido Test de Turing), 1951 (respuesta al debate sobre el tema de inteligencia en las máquinas, 1952 (estudio sobre morfogénesis), 1953 (estudio sobre el ajedrez) y 1954 es un regreso al tema tratado en 1936).

Al revisar el concepto de computabilidad “Afirmamos que la definición de computabilidad en Turing está en relación a la noción intuitiva de algoritmo, que trata sobre operar lógicamente dispositivos físicos... Si bien los conceptos utilizados por Turing son abstractos y corresponden a dispositivos que son ejecutados siguiendo reglas, en términos físicos, las instrucciones manejan automáticamente una máquina, en función del resultado deseado. (Salinas, 2011: 115). “Sostenemos como hipótesis que Turing realizó su investigación en el campo de la ciencia de la computación, específicamente en la relación entre máquina e instrucciones. Esta orientación se manifestó en las matemáticas, iniciándose en la solución negativa al problema denominado *Entscheidungsproblem*<sup>16</sup>, explicado en sus primeras publicaciones sobre funciones computables hasta sus estudios para entender el funcionamiento del cerebro del hombre”. (Salinas, 2011: 116)

Turing define su Automata, concepto matemático y abstracto de la computadora: “... en 1936... publica su trabajo en la Sociedad de Matemáticas de Londres, titulado: *On Computable Numbers, with application to the Entscheidungsproblem*, en el que define un dispositivo abstracto, que fue nombrado máquina de Turing. Éste es un concepto matemático basado en un procedimiento secuencial que se ejecuta en la máquina. La investigación de Turing concluye en la demostración de que no es posible obtener un procedimiento secuencial que calcule todas las proposiciones matemáticas, contradiciendo la propuesta hecha por Hilbert, quien sostuvo que las matemáticas son decidibles. Se supone que Turing conoció el artículo, publicado por Church en 1936, observando la equivalencia de los resultados con los que él había obtenido sobre el *Entscheidungsproblem*” (Salinas, 2011: 116)

En la teoría de autómatas<sup>17</sup>, “... la máquina de Turing es un autómata base, todos los demás tipos de autómatas se definen como deducidas de ella. La definición contiene la especificación de un procedimiento secuencial, que ejecuta instrucciones de

16 En ciencias de la computación y matemáticas, Entscheidungsproblem (problema de decisión) fue el reto en lógica simbólica de encontrar un algoritmo general que decidiese si una fórmula del cálculo de primer orden es un teorema. En 1936, de manera independiente, Alonzo Church y Alan Turing demostraron ambos que es imposible escribir tal algoritmo. Como consecuencia, es también imposible decidir con un algoritmo si ciertas frases concretas de la aritmética son ciertas o falsas.

17 Teoría de autómatas, es el estudio de las máquinas abstractas, conceptos matemáticos, que implícitamente asocian la idea de Algoritmo. Una utilización básica de los autómatas es en la Teoría de los Lenguajes de Programación, para obtener traductores o compiladores desde un programa en lenguaje fuente para lograr en programa en lenguaje objeto que será ejecutado por la computadora. Libro clásico de la teoría de compiladores Formal Languages and their Relation to Automata, por Hopcroft, J, Ullman, J, Hill, M, en <http://portal.acm.org/citation.cfm?id=1096945>. Sábado, 5 de abril 2008. 10:00 horas.

naturaleza elemental; también se le suele asociar la formulación del cálculo Lambda<sup>18</sup>. Turing en su tesis doctoral en la universidad de Princeton, trata sobre el sistema de lógica basado en ordinales. En esta investigación introducirá el concepto de Oráculo en la máquina, brillante concepto que permite encapsular conceptos sobre la complejidad en relación a los dispositivos automáticos” (Salinas, 2011: 117-118).

En nuestro documento, *Concepto de Computabilidad de Alan Turing*, indicamos que “La máquina de Turing suele ser descrita, como si fuera un dispositivo físico, que no es posible fabricarla debido a la definición abstracta de sus componentes. La máquina está constituida por una unidad de control (que determina el estado en que se encuentra la máquina y actúa sobre la memoria), una memoria que es una cinta de longitud infinita en ambos sentidos (izquierda y derecha), está contiene celdas una a continuación de otra (cada celda puede contener un símbolo).” (Salinas, 2022: 6)

Ratificamos con respecto a nuestra actualidad, “Hoy la máquina de Turing se representa mediante dispositivos mecánicos eléctricos, también suele ser mostrado en programas<sup>19</sup> de computadora, todos éstos trabajos son de interés académico. En el transcurso del tiempo se han presentado diversas definiciones para la máquina manteniéndose el sentido original.” (Salinas, 2011: 118)

Tomamos en cuenta lo enunciado por el filósofo Haugeland<sup>20</sup> sobre la máquina de Turing: “Una máquina de Turing consta de dos partes: una cabeza y una cinta. La cinta es sólo un medio de almacenamiento pasivo: a lo largo está dividida en casillas, cada una de las cuales puede contener un elemento. ... La cinta se usa también para las entradas y salidas, escribiendo elementos en ella antes que empiece la máquina, y leyendo los resultados después que para. La cabeza es la parte activa de la máquina que va, a saltitos, hacia atrás o hacia delante a lo largo de la cinta, casilla por casilla, y, mientras lee o escribe los elementos. En cada paso dado, la cabeza va “explorando”... También en cada paso, la propia cabeza se encuentra en estado interno.... este estado cambia de un paso

18 El cálculo lambda fue presentado en el capítulo anterior. este fue formulado por Alonzo Church con ayuda de C. Kleene, quienes describen funciones matemáticas.

19 Ver <http://www.microsiervos.com/archivo/ordenadores/premio-máquina-turing-si.html>. Sábado, 5 de abril de 2008, 18:00 horas.

20 John Haugeland, (nacimiento: Harrisburg, Illinois, 13 de marzo 1945; fallece: 23 de junio 2010) Profesor de filosofía de la Universidad de Chicago. Presidente del departamento de filosofía desde 2005 hasta 2007. Enseñó en la Universidad de Pittsburgh y la Universidad de California en Berkeley.

al siguiente. ... La cabeza entrará en un estado especial llamado “para”, en cuyo caso la máquina se detiene – dejando sus salidas en la cinta” (Haugeland, 1999:127-129)

A modo de conclusión, deducimos que “Turing conceptúa la máquina como un procedimiento que puede ser ejecutado por un hombre, sin ambigüedad, garantizando que se hará sólo lo que está especificado. Es una definición que está relacionada al operando de la mente humana, no decimos que corresponda a la totalidad de cómo opera la mente, pero hay parte de ésta en la especificación dada” (Salinas, 2011: 120), así mencionamos lo que dice Turing: “...Nosotros podemos comparar a un hombre en el proceso de cálculo de un número real a una máquina que sólo es capaz de un número finito de condiciones  $q_1, q_2, \dots, q_r$  que se llama m-configuraciones” (Turing, 1936: 2)

Encontramos en la lectura del libro de Von Neumann<sup>21</sup>, *The Computer and The Brain*, publicado póstumamente en 1958 reflexiones “como metáfora sobre la similitud de la computadora con el funcionamiento del cerebro. En el caso de las computadoras refiere a los códigos y su rol en el control del funcionamiento de una computadora. Precisa que estos códigos adquieren la forma de ser completos, en el sentido que permiten el funcionamiento de la máquina e indica que la máquina de Turing se refiere al concepto de código corto, como código elemental de la máquina. (Salinas, 2022: 7)

“El lógico inglés A. M Turing, mostró en 1937... que es posible desarrollar sistemas de códigos de instrucciones para una máquina computadora... Entonces el sistema de instrucciones que hizo una imitación de máquina como el comportamiento de otra es conocido como código corto” (von Neumann, 1958:71)

Resumiendo lo tratado sobre la máquina de Turing: “La observación que hace von Neumann sobre la máquina de Turing es que esta captura la esencia del código corto que hace que la máquina se comporte como si fuera otra máquina.” (Salinas, 2022: 126). La computadora en el instante en que está en ejecución, resulta ser, una unidad entre la máquina y el programa que contiene y ejecuta, ambas están

21 John von Neumann zu Margitta, (nacimiento: Budapest, 28 de diciembre de 1903; fallece: Washington DC, 8 de febrero de 1957) matemático húngaro-estadounidense. Recibió su doctorado en matemáticas de la Universidad de Budapest a los 23 años. Padre de la teoría de juegos, publicó *Theory of games and economic behavior* junto a Oskar Morgenstern, en 1944. Pionero de la computadora y de la aplicación de la teoría de operadores a la mecánica cuántica. La computadora EDVAC, desarrollada por Von Neumann, Eckert y Mauchly.

relacionadas como una unidad a la que se describimos como “máquina”.

#### IV. COMPLEJIDAD COMPUTACIONAL.

Podemos comprobar “en los textos de la ciencia de la computación que tratan sobre la complejidad muestran dos orientaciones de investigación, una relacionada a la cantidad de tiempo exigido en la computación de un programa, y otra al estudio del tamaño de un programa en cuanto a la cantidad de código requerido para resolver determinado problema a computar... En ambos casos (complejidad de tiempo y de espacio) fueron ‘ignorados’ por Turing, dado que consideramos que era consciente de estas dificultades, había que dejar de lado para lograr la precisión en la definición alcanzada, expresada en la abstracción de la máquina con cinta de memoria infinita y sin tiempo límite de ejecución.” (Salinas, 2011: 191-192). Aclarando que en la ejecución de un programa en la máquina de Turing, esta debe terminar el proceso, caso contrario diremos que no es computable, esto se conoce como el problema de la parada<sup>22</sup>.

Sobre el concepto de complejidad en el tiempo y el espacio, sostenemos: “En el caso, de la complejidad en el tiempo, se investiga la duración de la ejecución de un programa en una computadora, y como se afecta, cuando se trata de valores que aumentan en su argumento, considerando que una función recursiva es una definición de sí misma, por lo tanto, el tiempo requerido para calcular un valor es el tiempo para el cálculo del anterior argumento más un tiempo adicional. El comportamiento del tiempo para el cálculo mediante las funciones recursivas se comporta como una función polinómica<sup>23</sup>, por esta razón se les denomina problemas de tipo P, siempre que se ejecutan en una máquina determinista<sup>24</sup> de Turing, y se dice que son problemas del tipo NP, en los casos que se ejecutan en una máquina de Turing no determinístico<sup>25</sup>... En cuanto a la complejidad de espacio, se considera el

22 El problema de la parada o problema de la detención para máquinas de Turing indica que si una Máquina de Turing M y una secuencia de caracteres, si M termina su proceso en un número finito de pasos es computable, caso contrario es indecidible (no computable) y ninguna máquina de Turing lo puede resolver. Lo propuso en su artículo *On Computable Numbers with an Application to the Entscheidungsproblem* (1936)

23 Polinomios expresiones algebraicas de la forma  $P(x) = a^n x^n + a^{n-1} x^{n-1} + \dots + a^1 x + a^0$ .

24 Una máquina de Turing determinística, es aquella en la que la función de transformación de los estados, está precisamente definida, es decir cuando la máquina de Turing está en un estado y lee un símbolo de la cinta, pasa a otro estado y se desplaza en la cinta.

25 Una máquina de Turing determinística es aquella en la que la función de transformación de los estados se da el caso que se puede pasar a un estado u otro para una situación determinada.

funcionamiento de un programa en la computadora en relación a los recursos de memoria, esta forma de plantear el problema de estudio sobre el comportamiento de la duración de ejecución en relación al tamaño, se dice que son de una complejidad del tipo L, que está incluida en los problemas tipo P, y en el mismo sentido con la complejidad de espacio no determinístico, denominado del tipo NL.” (Salinas, 2011: 192-193)

También encontramos “Otra forma de plantear el caso de la complejidad computacional, es estudiando la variable del tamaño del programa de computadora, desligándose en cierto sentido del concepto de la recursividad, incorporando en la investigación el tamaño de memoria requerida para el funcionamiento de un programa, en esta orientación encontramos los estudios de Kolmogorov<sup>26</sup>.” (Salinas, 2011, 193), así lo sostiene “Mosterín refiere a la complejidad de Kolmogorov, en el sentido de que la define en relación al tamaño de lo que se está describiendo, incluyendo en la definición la regularidad hasta su ausencia, de tal forma que el resultando es más simple en una descripción de una secuencia de caracteres cuando hay regularidad, ocurriendo lo contrario cuando no hay regularidad en lo que se describe.” (Salinas, 2011: 193)

“En 1965 Andrei Kolmogorov introdujo de un modo preciso la noción de complejidad - llamada actualmente en su honor complejidad de Kolmogorov o K - como una medida de la información individual o aleatoriedad de las secuencias” (Mosterín y Torretti, 2002:97).

El filósofo y matemático Gregori Chaitin<sup>27</sup>, “afirma que Turing descubre la no computabilidad y refiere a que Hilbert insistió en la existencia de un procedimiento mecánico que decidiese si la demostración se ajusta a reglas especificadas, Hilbert nunca aclaró que significó el procedimiento mecánico” y continua Chaitin. “En este sentido se expresa la complejidad, en la medida de precisar el procedimiento, que contenga aspectos aun no estudiados a profundidad, como el de la aleatoriedad y la no computabilidad... Chaitin indica sobre al problema de complejidad de tamaño, no le importa cuán rápido se efectúe un programa. Explica que la complejidad se encuentra en relación a la cantidad de

26 Andrei Kolmogorov, (nacimiento: Moscú, 25 de abril de 1903; fallece: 20 de octubre de 1987), matemático ruso. Obtuvo su doctorado en la universidad estatal de Moscú en 1929, hizo importantes progresos en topología, estadística, fundador de la teoría de la complejidad algorítmica.

27 Gregori J.Chaitin, (nacimiento: Chicago, 24 junio de 1947). De padres argentinos. Estudia matemática en la universidad de Buenos Aires. Trabaja en el centro de investigación J. Watson de IBM en York Town, Heights New York. Trabaja en la teoría algorítmica. Ha publicado diversos artículos y libros sobre temas de la ciencia de la computación.

información que está contenida en las instrucciones y datos que el programa efectuará. (Salinas, 2011: 193-194)

“Es una teoría del largo del programa, del tamaño del programa, no del tiempo de su corrida; no importa si es un programa rápido o no. Se mide por la complejidad de H de una tarea de cómputo por la cantidad de información que es necesario tener en el programa. La mínima cantidad de información que se requiere” (Chaitin, 2002: 56).

Von Neumann, en los años 1950, resalta la importancia de la complejidad temporal de los cálculos, y visualiza la necesidad de mejorar la potencia de las computadoras; potencia que a la fecha se ha multiplicado por varios números en relación a disminuir el tiempo de ejecución de las computadoras, implicando diversos aspectos que son parte de estas, tales como: memoria, procesador, circuitos de transferencia de datos entre otros.

Para el estudio de la complejidad, se requiere el concepto de la aleatoriedad, así lo indicamos: “Dentro del concepto de complejidad de la computación encontramos el caso de la aleatoriedad. Ésta se manifiesta en los diversos programas como el de la generación de números aleatorios, utilizados en diversos procesos automáticos, y sucede que en su mayoría son pseudogeneradores, porque estos programas empiezan con un valor que suele llamarse “semilla” y a partir de ésta se obtienen los siguientes números. Se define un intervalo de números, cuando se repite el valor de la semilla. Una solución al caso es implementada mediante un componente físico de generación del número aleatorio que es un circuito basado en el ruido que produce un diodo (este mecanismo tiene la naturaleza de ser aleatorio).” (Salinas, 2011: 194)

Regresado a Chaitin, sostiene “... que Turing y Church definen cada uno un lenguaje de programación: Turing utiliza un lenguaje de máquina, mientras que Church utiliza un lenguaje de alto nivel comparado al lenguaje LISP en relación al cálculo lambda de las funciones recursivas” (Salinas, 2011, 194), indica: “El de Turing no es un lenguaje de alto nivel, como el LISP; se trata más bien de lenguaje de máquina. El código está formado por unos y ceros que se le suministra al procesador central de un ordenador” (Chaitin, 2003: 32).

En el caso de la Teoría de Turing, encontramos la definición de la máquina de Turing con Oráculo, trabajo presentado en 1938, como “... forma de plantear la complejidad de la computación... se han conjeturado dos opciones: Una que dice que

es posible construir una computadora con el dispositivo de Oráculo con una nueva física, así lo dice Roger Penrose<sup>28</sup>, otra interpretación refiere a que la máquina de Turing ya contiene la capacidad de cómputo suficiente y en la medida que comprendamos situaciones sobre sus procedimientos y variables involucradas, éstas podrán ser expresadas en procedimientos, en este sentido lo expresa Stephen Hawking<sup>29</sup>, indica que si ya simulamos en una computadora el funcionamiento del cerebro de una lombriz no hay ninguna diferencia para lograr simular el funcionamiento de un cerebro humano<sup>30</sup> (Salinas, 2011: 195-196)

Así, a modo de conclusión, la Teoría de Turing se encuentra en relación a “estudios que busca el entendimiento del funcionamiento del cerebro humano, considerando que en la medida que comprendamos sobre su funcionamiento, se podrá formular algoritmos que permitan simular diversos aspectos del cerebro humano.” (Salinas, 2011: 196)

## V. CONCLUSION.

Según lo tratado, solemos manejar el concepto de Modelo en dos sentidos, tal como fueron expuestos, como representación y como representante, en cierto sentido como imagen u objeto. Resumiendo el Modelo como una Teoría o realización de esta.

Consideramos que lo representado como Modelo, es el objeto que sirve de muestra, así podemos afirmar que la máquina de Turing es el modelo elemental de cualquier dispositivo mecánico eléctrico que compute, “al ser una máquina de lápiz y papel, según definición de Turing, define las características de lo que se entiende por computable en relación al formalismo establecido en los dispositivos de la

28 Roger Penrose, (nacimiento: Colchester, Reino Unido, 8 de agosto de 1931). Físico matemático, profesor emérito de matemáticas en la Universidad de Oxford. miembro de la Royal Society de Londres en 1972, ganó el Science Book Prize en 1990, y compartió el Premio Wolf en Física con Stephen Hawking en 1988. Sus publicaciones son diversas: *Techniques of Differential Topology in Relativity* (1973), *Spinors and Space-Time* (junto con Wolfgang Rindler; vol. 1, *Two-Spinor Calculus and Relativistic Fields*, 1984; vol. 2, *Spinor and Twistor Methods in Space-Time Geometry*, 1986), *La nueva mente del emperador* (1989), *Las sombras de la mente* (1994), *Lo grande, lo pequeño y la mente humana* (1997), y su último título, *El camino a la realidad* (2004).

29 Stephen William Hawking, (nacimiento: Oxford, 8 de enero de 1942; fallece: Cambridge, 14 de marzo de 2018). Físico miembro de la Real Sociedad de Londres. Titular de la cátedra Lucasiana de Matemáticas en la universidad de Cambridge. Se doctora en física en 1966 en Oxford, en 1979 es nombrado catedrático lucasiano. Investigador, ideó nuevas técnicas matemáticas en estudios de la relatividad, los agujeros negros, descubriendo que estos emiten radiación. También sustenta que después del Big bang, se crean objetos supermasivos.

30 Lo mencionado por Hawking se encuentra en el libro *Lo grande lo pequeño y la mente humana*, en la que resultar compartir la autoría con Penrose Roger, Shimony Abner, y Cartwright Nancy

máquina abstracta y su relación lógica mediante el procedimiento efectivo. (Salinas, 2011: 214-215)

Consideramos que la representación como Modelo de lo representado, resulta la “... Teoría que deriva modelos explicativos. La máquina de Turing siendo la Teoría y los autómatas que resultan ser Modelos derivados de la combinación de máquinas de Turing.” (Salinas, 2011: 215)

Sostenemos, según el punto de vista que pretendamos priorizar, la máquina de Turing es un Modelo básico de la computadora, o es la Teoría. La máquina de Turing como Teoría nos permite concebir otros modelos de autómatas, más elaborados y complejos, pero como Modelo es una realización de la Teoría de Autómatas.

Afirmamos que “El concepto de computabilidad está en la base de lo que expresa el procedimiento efectivo, este no solo es la secuencia de instrucciones, también incorpora los dispositivos que constituyen la máquina y la necesaria comunicación entre estos.” (Salinas, 2011: 215)

Consideramos oportuno incluir cuando “... se refiere a la correspondencia de la máquina de Turing, como una máquina de lápiz y papel, que ejecuta un procedimiento efectivo en el sentido de lo que puede hacer un hombre siguiendo instrucciones sin entendimiento que el necesario para ejecutar, los programas y las computadoras se adecuan a este tipo de realización de la abstracción mencionada. El Modelo como formulación aplicada a una realidad, es donde se ejecutará el programa, así en su utilización en las computadoras en situaciones específicas, determinará la realización del modelo. (Salinas, 2011: 215-216)

Entendemos que el elucidar el término de Modelo en relación a lo representado y su representación, “contribuye a la interpretación sobre aspectos cognitivos en la construcción de programas para ser utilizados en computadoras. Para iniciar la construcción de un programa se requiere definir los diversos componentes a ser expresados en reglas, para este fin, se suele construir un Modelo denominado prototipo (no es el programa, pero nos dice mucho sobre el funcionamiento del programa) de forma que explique las formas y cómo va a funcionar. Sirve como herramienta para la aceptación de lo que se hará o se construirá mediante una codificación.” (Salinas, 2011: 61-62)

Así también, “... el concepto de Modelo, cuando se trata de un programa de computadora que está funcionando en una organización, entonces el programa

es un Modelo de un sistema en funcionamiento, que puede ser utilizado en otra organización que requiera el programa en sentido similar en donde ya está funcionando.” (Salinas, 2011: 63)

Resaltamos el hecho de que el concepto de computabilidad “... es más amplio al expresado mediante la función recursiva, entendemos que la utilización de este tipo de función matemática en el inicio de la ciencia de la computación, inspiró y posibilitó la construcción de una teoría de la computación, fue a través de las funciones recursivas donde la teoría alcanza su mejor expresión. Se interpretó como equivalentes y coextensivas la función efectiva y el procedimiento efectivo, en relación a la función computable... El procedimiento efectivo es la mejor expresión que elucida el algoritmo, como procedimiento automático, en cierto sentido relacionado a la axiomatización y deducción de teoremas de una teoría formalizada.” (Salinas, 2022: 19)

Según lo expresa Jack Copeland<sup>31</sup>, autoridad sobre temas de Alan Turing, “... en su artículo publicado en el 2004, en el libro editado por Luciano Floridi<sup>32</sup>, *Philosophy of Computing and Information*, expresa sobre la importancia de la máquina de Turing con respecto al concepto de programa almacenado” (Salinas, 2022: 19), así nos dice: “El principio básico de la computadora moderna, es la idea de controlar las operaciones de la máquina por medio de un programa de instrucciones de códigos almacenados en la memoria de la computadora que fue pensado por Alan Turing en 1935” (Copeland, 2004:3)

De manera que al tratar sobre la computabilidad, estamos tratando también del concepto de algoritmo, entendido como un procedimiento que tiene un objetivo, el calcular, o mejor dicho computar una respuesta a un problema, utilizando un proceso, conjunto de pasos a ser ejecutados, que son previamente especificados en detalles necesarios y suficientes. “Las instrucciones son expresadas mediante un texto finito, y no es posible la actividad creadora en el momento de ejecución.” (Salinas, 2011: 188)

“Si bien el concepto de procedimiento efectivo es matemático, resulta que los matemáticos cuando

31 Jack Copeland (nacimiento: United Kingdom, 1950) es profesor de filosofía en la Universidad de Canterbury (Nueva Zelanda). Recibió su D.Phil y B.Phil en filosofía de la Universidad de Oxford en 1979 por sus investigaciones sobre lógica modal y no-clásica. Es director del Archivo Turing para la Historia de la computación en Canterbury (Nueva Zelanda) desde 1985. Es considerado un experto en los temas de Alan Turing.

32 Luciano Floridi, (nacimiento: Roma, 16 de noviembre de 1964), filósofo italiano, conocido por sus trabajos de filosofía de la información y la ética informacional.

defienden un procedimiento, este no es necesariamente riguroso en cuanto a la precisión, se requiere de explicación, y si se habla de reglas estas no indican el orden en que deben ser ejecutadas. La máquina de Turing define un proceso, al que llamamos procedimiento efectivo, que contiene instrucciones elementales y precisas, de forma que resulta ser un programa, tal como hoy son los que se ejecutan en las computadoras, pero en Turing es un código elemental.” (Salinas, 2011: 188)

Hans Hermes<sup>33</sup> nos dice sobre el concepto de algoritmo: “Del concepto de algoritmo podemos extraer toda una serie de importantes conceptos ulteriores. Algunos de esos conceptos son los de computabilidad, enumerabilidad y generalidad” (Hermes Hans, 1984: 32), y nosotros resaltamos la relación entre computabilidad y algoritmo.

Indicamos como conclusión sobre el concepto de computabilidad en Turing que “no precisó sobre el mecanismo del funcionamiento de los componentes de su máquina, en su documento de 1936, refirió a su máquina como una definición matemática, y en su documento de 1938, le incorporó un dispositivo al que denominó oráculo, que resulta ser una máquina con más potencia, que tiene la capacidad de saber si un problema es o no computable. En este tema la computabilidad incorpora aspectos que no se explican en las funciones recursivas generales, así mismo deja espacio a la comprensión de la complejidad de la computabilidad, que incluye aspectos mencionados como complejidad de tiempo y de espacio en el uso de las computadoras y aspectos como tamaño del programa o interrelación de comunicación y proceso entre computadoras.” (Salinas, 2011: 191)

Consideramos pertinente incluir sobre los periodos de tiempo en la historia de la computabilidad, mencionado por Robert Soare: “El período moderno de la teoría de computabilidad se puede dividir en tres períodos. 1. Era de definibilidad Lambda 1931-1935. 2. Era de la teoría de la Recursión 1935-1995. 3. Era de la Computabilidad 1996-al presente” (Soare Robert, 2007: 1)

Sabemos, y ya no hay discusión sobre la importancia de la Teoría de Turing, específicamente su Modelo abstracto de máquina, “comprendemos que esta permitió la fabricación de una computadora,

33 Hans Hermes, (nacimiento: Neunkichen, 12 de febrero de 1912; fallece: 10 de noviembre del 2003), matemático, física, química, biología y filosofía, profeso en Freiburg y Munich, estudioso de Frege, profesor de Mosterin en la universidad de Münster

así tomamos lo mencionado por Martin Davis<sup>34</sup> le otorga un papel importante a la definición de máquina de Turing.” (Salinas, 2011: 218)

“Alan Turing descubre lo universal (o de uso general) de la computadora digital como una abstracción matemática... este trabajo abstracto ayudó a Turing y John von Neumann hacia la concepción moderna de la computadora electrónica” (Martin Davis, 2006:125)

La arquitectura de la computadora de von Neumann, fue formulada sobre la base del trabajo de Turing, así lo manifiesta Martin Davis “Prácticamente todos los ordenadores de hoy de más de \$ 10 millones de las súper computadoras hasta los chips de teléfonos celulares y los furbies<sup>35</sup>, tienen una cosa en común: todos ellos son "máquinas de Von Neumann", variaciones sobre la arquitectura básica de John von Neumann, construidas sobre la base del trabajo de Alan Turing, establecido en 1940” (Martin Davis, 2006:126)

“Sabemos que von Neumann describe al final de su libro *The Computer and the Brain* que el lenguaje del cerebro no es el lenguaje de las matemáticas, indica de que hay un lenguaje en relación al código que es más sencillo al utilizado para hablar, dado que el procedimiento efectivo en la máquina de Turing está constituido por códigos básicos necesarios y suficientes para operar los diversos dispositivos, basado en la definición abstracta de autómatas.” (Salinas, 2011: 222), Von Neumann lo expresa de esta forma: “... debe tenerse en cuenta que el lenguaje aquí involucrado puede corresponder a un tipo de código en el sentido descrito anteriormente en lugar de un código completo: cuando hablamos de las matemáticas, podemos estar hablando una lengua secundaria, basada en la lengua primaria verdaderamente utilizados por el sistema nervioso central” (von Neumann, 1958:82)

Finalmente resaltamos la importancia en diferenciar las ideas presentadas en los diversos niveles que hemos tratados, especialmente para referirnos a los conceptos de Modelo y Teoría, dado que nos ayudan a un mayor entendimiento sobre el concepto de computabilidad; el ignorar estas diferencias, no ayudan en mucho para la explicación

34 Martin Davis, (nacimiento: Nueva York, 1928; fallece: California, 1 de enero de 2023) matemático estadounidense conocido por su trabajo relacionado con el décimo problema de Hilbert. Obtuvo su PhD en la Universidad de Princeton en 1950 y su tutor fue Alonzo Church. Es profesor emérito de la Universidad de Nueva York. Es coinventor del algoritmo de Davis-Putnam y de los algoritmos DPPL. También es coautor, junto con Ron Sigal y Elaine J. Weyuker de *Computability, Complexity, and Languages, Second Edition: Fundamentals of Theoretical Computer Science*

35 Furbies, juguetes electrónicos para niñas y niños inclusive para jóvenes.

de temas que suelen estar presentes en la ciencia de la computación.

La reflexión crítica sobre Modelo y Teoría, sus diferencias, campos de acción, y las teorías que la incluyen, son parte para una filosofía de la computación que nos ayudará a comprender el concepto de Computabilidad, ampliando nuestro alcance cognitivo y uso de esta. Aceptamos que nuestro punto de partida incluye las reflexiones de lo tratado en la tradición de la filosofía y en la epistemología<sup>36</sup> en particular sobre los temas técnicos y científicos propios de nuestra disciplina de la computación.

## VI. REFERENCIAS.

- [1] Calvo Vélez, David. (2006). *Modelos teóricos y representación del conocimiento*. Tesis de Grado de Doctor. Universidad Complutense de Madrid, Facultad de Filosofía. Madrid, España.
- [2] Chaitin Gregori (2002). *Información y Azar*. Boletín de la Asociación Matemática Venezolana, Volumen IX, Número 1, PP 55-81.
- [3] \_\_\_\_\_ (2003). *Ordenadores, Paradojas y Fundamentos de las Matemáticas*, [www.cs.auckland.ac.nz/~chaitin/investigacion-y-ciencia.pdf](http://www.cs.auckland.ac.nz/~chaitin/investigacion-y-ciencia.pdf). (2009, 12 de abril).
- [4] Ferrater Mora. (2004). *Diccionario de Filosofía*. Barcelona, España: Editorial Ariel S.A.
- [5] Floridi, Luciano. (1999). *Philosophy and Computing: An introduction*. New York, USA: Routledge of the Taylor & Francis Group
- [6] \_\_\_\_\_ (2004). *Philosophy of Computing and Information*. London, Kingdom: Blackwell Publishing Ltd.
- [7] \_\_\_\_\_ (2007). *Semantic Conceptions of Information*. [plato.stanford.edu/archives/spr2007/entries/information-semantic](http://plato.stanford.edu/archives/spr2007/entries/information-semantic). (2010, 20 de enero)
- [8] Haugeland, John. (1999). *La inteligencia Artificial*. (Tulli Irene Trad.). Buenos Aires, Argentina: Siglo XXI.
- [9] Hermes Hans. (1984). *Introducción a la teoría de la computabilidad Algoritmos y Máquinas*. (Garrido Manuel y Martin Aránzazu trad.). Madrid, España. Editorial Tecnos.
- [10] Hilbert David. (1900). *Mathematical Problems Lecture delivered before the International Congress of mathematical at Paris 1900*.

36 Epistemología, del griego ἐπιστήμη episteme (conocimiento) y λόγος logos (estudio), es parte de la filosofía que trata sobre el conocimiento científico, las teorías, conceptos, y fundamentos.

- aleph0.clarchku.edu/~djoyce/hilbert/problems.html. (2008, 18 de marzo).
- [11] Kolmogorov A.N. y Fomin S.V. (1975). *Elementos de la Teoría de Funciones y del Análisis Funcional*. (Vega Carlos Trad.). Moscú, URSS: Editorial MIR.
- [12] Manzano María. (1989). *Teoría de Modelos*. Madrid, España: Alianza Editorial.
- [13] \_\_\_\_\_ (2005). *¿Qué es esa cosa llamada lógica?* En Representación y Lógica. [be.geocities.com/fapespcleime/resources/conferencias/simposiorepresentacionylogicidad2005.pdf](http://be.geocities.com/fapespcleime/resources/conferencias/simposiorepresentacionylogicidad2005.pdf) (2008, 15 de mayo)
- [14] \_\_\_\_\_ (2006). *Sobre el Razonamiento Formal*. En 50 Años de la Inteligencia Artificial. (Caballero Antonio, Miguel Sergio Eds). Albacete, España: Universidad de Castilla-La Mancha, Departamento de Sistemas Informáticos.
- [15] Martin Davis. (1987). *Influencias de la Lógica Matemática en las Ciencias de la Computación*. (García Facundo Trad.). [www.econ.uba.ar/www/departamentos/humanidades/plan97/logica/Legris/apuntes/Davis.pdf](http://www.econ.uba.ar/www/departamentos/humanidades/plan97/logica/Legris/apuntes/Davis.pdf). (2009, 15 de julio).
- [16] \_\_\_\_\_ (2006). *The Church–Turing thesis consensus and opposition*. [citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.91.7093](http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.91.7093). (2009, 12 de enero).
- [17] \_\_\_\_\_ (2006). *What is Turing Reducibility?* Notices of the AMS, Volume 53, Number 10, November, PP 1218-1219.
- [18] \_\_\_\_\_ (2008). *Hilbert's Tenth Problem is Unsolvable*. [Naughtybag.files.wordpress.com/2008/03/martindavis.pdf](http://Naughtybag.files.wordpress.com/2008/03/martindavis.pdf). (2009, 13 de mayo).
- [19] Mosterín (2008). *Conceptos y Teorías en la Ciencia* (1a ed.). Madrid, España: Alianza Editorial S.A.
- [20] Mosterín Jesús, Torretti Roberto. (2002). *Diccionario de lógica y filosofía de la ciencia*. (1a ed.). Madrid, España: Alianza Editorial S.A.
- [21] Penrose Roger, Shimony Abner, Cartwright Nancy, Hawking Stephen. (1999). *Lo grande lo pequeño y la mente humana*. (1a ed.) (García Javier Sanz Trad.). Madrid, España: Cambridge University Press.
- [22] Salinas Molina Miguel. (2011). *Computabilidad y máquina de Turing*. Tesis de Grado de Magister en Filosofía con Mención en Epistemología. Universidad Nacional de San Marcos. Lima Perú.
- [23] \_\_\_\_\_ (2021). *Reflexiones sobre el significado de información*. Revista de Investigación de sistemas e informática. Volumen 14 Numero 2: 67-75 (2021).
- [24] \_\_\_\_\_ (2022). *La Probabilidad y la Complejidad: Un enfoque Epistemológico*. Revista de Investigación de sistemas e informática. Volumen 15 Número 1: 101-112.
- [25] \_\_\_\_\_ (2022). *Concepto de Computabilidad en Alan Turing*. Revista de Investigación de sistemas e informática. Volumen 15 Numero 2: 87-105.
- [26] Soare Robert. (1996). *Computability and Recursion*. Bulletin of Symbolic Logic, Volume 2, Number 3, PP 284-321.
- [27] \_\_\_\_\_ (2007). *Computability and Incomputability*. [www.people.cs.uchicago.edu/~soare/History/siena.pdf](http://www.people.cs.uchicago.edu/~soare/History/siena.pdf). (2009, 15 de febrero)
- [28] \_\_\_\_\_ (2009) *Turing Oracle Machine, Online computing and three Displacements in Computability Theory*. [www.people.cs.uchicago.edu/~soare/History/turing.pdf](http://www.people.cs.uchicago.edu/~soare/History/turing.pdf). (2009, 14 de febrero)
- [29] Turing, A. (1936). *On computable numbers, with application to the Entscheidungsproblem*. [www.thocp.net/biographies/papers/turing\\_oncomputablenumbers\\_1936.pdf](http://www.thocp.net/biographies/papers/turing_oncomputablenumbers_1936.pdf). (2008, 14 de junio).
- [30] \_\_\_\_\_ (1938). *Systems of logic based on ordinals* [www.turingarchive.org/browse.php/B/15](http://www.turingarchive.org/browse.php/B/15). (2008, 14 de junio).
- [31] \_\_\_\_\_ (1948). *Intelligent Machinery*: [www.alanturing.net/turing\\_archive/archive/1/132/L32-001.html](http://www.alanturing.net/turing_archive/archive/1/132/L32-001.html). (2008, 14 de junio).
- [32] \_\_\_\_\_ (1950). *Computing machinery and intelligence* [www.loebner.net/Prizef/TuringArticle.html](http://www.loebner.net/Prizef/TuringArticle.html). (2008, 14 de junio).
- [33] \_\_\_\_\_ (1951). *Can Digital Computer Think* [www.turingarchive.org/browse.php/B/5](http://www.turingarchive.org/browse.php/B/5). (2008, 14 de junio).
- [34] \_\_\_\_\_ (1952). *The Chemical Basis of Morphogenesis* [www.dna.caltech.edu/courses/cs191/paperscs191/turing.pdf](http://www.dna.caltech.edu/courses/cs191/paperscs191/turing.pdf). (2008, 14 de junio).

- [35] \_\_\_\_\_ (1953). *The Chess*. <http://www.turingarchive.org/browse.php/B/7> (2009, 20 de julio).
- [36] \_\_\_\_\_ (1954). *Solvable and Unsolvable Problems*. <http://www.turingarchive.org/browse.php/C/24>. (2010, 29 de julio)
- [37] Torretti Roberto. (1998). *El Paraíso de Cantor: La Tradición Conjuntista en la Filosofía de las Matemáticas*. Santiago, Chile: Editorial Universitaria, Universidad Nacional Andres Bello.
- [38] Von Neumann John. (1958). *The Computer and the Brain*. (1a ed.) Yale USA: Yale University Press.
- [39] Wittgenstein Ludwig. (1978). *Observaciones sobre los fundamentos de la matemática*. (Reguera Isidoro Trad.). Madrid, España: Alianza Editorial.