Un Siglo de Controversia Sobre los Fundamentos de la Matemática

Categoría	Documento
Estatus	Terminado
Modificado	@5 de junio de 2024 18:41
Creado	@28 de febrero de 2024 16:48
Tipo	Investigación / Trabajo

Algunas personas veıan la teorıa de conjuntos como a una enfermedad.

Poincar é el gran matemático Franc és dijo la teoría de conjuntos es una en fermedad, de la cual espero se recuperaran las generaciones futuras. Pero otras

personas rehicieron toda la matemática usando el enfoque de la teorıa de con- juntos, con preguntas generalizadas.

Poincaré, el renombrado matemático francés, consideraba la teoría de conjuntos como una enfermedad de la que esperaba que las futuras generaciones se recuperaran. Sin embargo, otros matemáticos redefinieron por completo la matemática utilizando un enfoque basado en la teoría de conjuntos.

David Hilbert era un matem ático muy importante alrededor del cambio de siglo. No como a Poincar é, un matem ático Franc és muy importante – Hilbert era un matem ático Alem án muy importante – a Hilbert le gustaba la teor ía de conjuntos. A el le gustaba este enfoque abstracto Cantoriano. Y Hilbert tuvo la idea de resolver de una vez y por todas estos problemas. Como iba a hacerlo?

La forma en que Hilbert iba a hacerlo es usando el m étodo axiom ático, el cual por supuesto viene desde Euclides – Hilbert no invent ó esto. Pero el hizo un significativo paso adelante.

David Hilbert, un matemático alemán importante, apreciaba la teoría de conjuntos y el enfoque abstracto de Cantor. Hilbert propuso resolver los problemas matemáticos utilizando el método axiomático, un enfoque que, aunque se remonta a Euclides, fue significativamente avanzado por él.

Hilbert dijo usemos toda la tecnolog ía de la l ógica simb ólica, la cual un mont ón de gente esta inventando, y vamos a un extremo final. Porque una de las razones por las que usted se mete en problemas y obtiene contradicciones en
matem ática con la teor ía de conjuntos es porque las palabras son muy vagas. Lo
que deseamos hacer para deshacernos de todos estos problemas en la matem ática
y en el razonamiento es deshacernos de los pronombres por ejemplo, usted no sabe a que se refieren los pronombres. Y hay toda clase de cosas que son vagas
en el lenguaje natural.

Hilbert dijo que la forma de deshacernos de todos esos problemas es desarrollar un conjunto finito de axiomas y un lenguaje artificial para hacer matem ática

– esta es la idea del formalismo llevada al l ímite.

Hilbert propuso utilizar la lógica simbólica para abordar los problemas y contradicciones en matemáticas debido a la vaguedad del lenguaje natural. Sugería desarrollar un conjunto finito de axiomas y un lenguaje artificial para la matemática, llevando el formalismo al límite.

El mundo real que conocemos es un absoluto desorden – cierto? – todo es complicado y enredado. Pero el único lugar donde las cosas deber ían ser absolutamente claras, blanco o negro, es en la matem ática pura.
Esta es la clase de afirmaci ón que Hilbert estaba haciendo, y el propuso esto como una meta, tener esta formalizaci ón de toda la matem ática y eliminar todos

los problemas. Ahora esto era un programa, no era algo que usted har ía en un fin de semana. Hilbert propuso esto como una meta para poner a la matem ´ ática
en una base firme. El y un grupo de colaboradores muy brillantes, incluyendo a John von Neumann, empezaron a trabajar en esto, y por un lapso de treinta a ños, lucia alentador. Y entonces – este es un r ápido resumen de un siglo de trabajo – entonces estoy seguro que todos ustedes saben que hab ían unos pocos
peque ños problemas!

Esto era la clase de cosa
que la l ógica matem ática estaba tratando de hacer, la idea de descomponer las
demostraciones en pasos cada vez mas peque ños.

Hilbert propuso la meta de formalizar toda la matemática y eliminar todos los problemas, afirmando que la matemática pura debería ser un lugar de absoluta claridad. Este programa extenso, trabajado por Hilbert y otros colaboradores brillantes, buscaba poner la matemática en una base firme, y parecía prometedor durante unos 30 años. La lógica matemática buscaba descomponer las demostraciones en pasos cada vez más pequeños.

Turing tiene que inventar el computador, porque Hilbert dice que deber ía haber una procedimiento mec ánico para decidir si una demostraci ón es correcta
o no. Turing dice que lo que Hilbert realmente quiere decir es que deber ía haber un programa de computador para chequear demostraciones. Pero primero
Turing tiene que decir lo que es un computador, es una m áquina de Turing, y todo esto en un articulo de Turing de 1936, cuando no hab ían computadores, as í que es una pieza fant ástica de trabajo. Y me gustar ía afirmar que esta es la
invenci ón del computador. Estos eran computadores de prop ósito general, esa
era la idea sobre el papel.

Turing inventó la computadora para cumplir con la propuesta de Hilbert de tener un procedimiento mecánico para decidir si una demostración es correcta o no. Turing interpretó esto como la necesidad de un programa de computadora para comprobar las demostraciones. Antes de poder hacer esto, Turing tuvo que definir lo que era una computadora, dando lugar a la máquina de Turing. Todo esto se detalla en un artículo de Turing de 1936, antes de la existencia de las computadoras, lo que se considera una asombrosa pieza de trabajo y la invención de la computadora.

La mec ánica cu ántica es la teor ía mas exitosa hasta ahora. Obtenemos transistores y computadores de ella. Pero incluso aunque Einstein ayud ó a contribuir a la creaci ón de la mec ánica cu ántica el la odiaba. Entonces parece que Einstein estaba equivocado. Dios juega a los dados!

La mecánica cuántica, de la cual derivan los transistores y las computadoras, es la teoría más exitosa hasta la fecha. A pesar de que Einstein contribuyó a su creación, la odiaba, sugiriendo que Dios juega a los dados.

En otras
palabras, quiz ás algunas veces la raz ón por la que usted no puede probar algo
no es porque usted es est úpido o no lo ha trabajado lo suficiente, la raz ón por la cual usted no puede probar algo es porque no hay nada ah í! Algunas veces la
raz ón por la cual usted no puede resolver un problema matem ático no es porque
usted no es lo suficientemente listo, o porque no es lo suficientemente resuelto,

es porque no hay soluci ón porque quiz ás la pregunta matem ática no tiene estructura, quiz ás la respuesta no tiene patr ón, quiz ás no hay orden o estructura

que usted pueda tratar de entender en el mundo de la matem ática pura.

Esta es una clase de noci ón l ógica de aleatoriedad en vez de una noci ón estad ística. No es como en f ísica donde usted dice que un proceso f ísico es aleatorio como el lance de monedas. No me importa de donde vienen las cosas.
Solo miro algo y dijo si tiene estructura o patr ón o no. Esta es una aleatoriedad

l ógica o estructural contraria a la imprevisibilidad y aleatoriedad f ísica. Es diferente - est án relacionadas muy cercanamente, pero son diferentes.

El concepto b ásico es mirar al tama ño del programa mas conciso, el menor programa - no me importa su tiempo de ejecuci ón - es el programa mas conciso
que calcula algo. Este es el n úmero de bits que tengo que darle al computador a fin de producir este objeto. Esa es mi descripci ón algor ítmica mas concisa de
algo, y as í es como mido su complejidad, su contenido de informaci ón algor ´ ítmica
o la complejidad del tama ño de su programa.

Ω = probabilidad de detenci ón

Como se define este n úmero? Es muy simple. Turing dijo que usted no puede decidir si un programa se detiene, no hay procedimiento mecanico para hacer eso. Y yo digo, considere un n úmero real Ω que es la probabilidad de que
un programa generado lanzando una moneda al aire se detenga. Entonces estoy
promediando sobre el problema de detenci ón de Turing, diciendo que si genero
un programa lanzando monedas al aire, cual es la probabilidad de que se detenga,
sin limite de tiempo? Entonces esto me dar á un n úmero real determinado si usted me dice - hay un sub índice - cual es el lenguaje de programaci ón.

¿Que es esto de ser m áximamente incognoscible? Bien, son los d ígitos o bits

de este n úmero. Una vez que fijo el lenguaje de programaci ón esta probabilidad de detenci ón es un n úmero real especifico, que depende de la escogencia

del computador, o el lenguaje de programaci ón en el cual genero un programa

lanzando monedas al aire. As í esto se hace un n úmero real especifico, y digamos que lo escribo en binario, as í que obtengo una secuencia de 1’s y 0’s, es

una definici ón muy simple. Bien, resulta que esos 0's y 1's no tienen estructura matem ática. No pueden ser comprimidos. Para calcular los primeros N bits de este n úmero se requiere un programa de N bits. Para ser capaz de probar lo que
son los primeros N bits de este n úmero se requieren N bits de axiomas. Esta es informaci ón matem áticamente irreducible, esa es la idea clave.

Entonces cual es la definici ón precisa de Ω? Genere un programa lanzando una moneda por cada bit, esas son lanzadas independientes de una moneda no cargada. El punto clave es que el programa tiene que ser "auto-delimitado".
El computador tiene que preguntar por cada bit uno por uno. Cada vez que el computador dice que desea otro bit del programa, usted lanza la moneda.
Y el computador tiene que decidir por si mismo si tiene suficientes bits, que tiene el programa completo. El programa tiene que ser auto-delimitante para definir esta medida de probabilidad correctamente.

La idea de Hilbert de ir al limite, de la formalizaci ón completa, y por razones epistemol ógicas, fue una controversia sobre los fundamentos de la matem ´ ática

- hay fundamentos? Y en una forma el proyecto fall ó, como he explicado, por
  el trabajo de G ödel y Turing. Pero aqu í estamos con estas formalizaciones completas que son los lenguajes de programaci ón de computadores, est ´ án en
  toda parte! Pagan mi salario, probablemente paguen su salario . . . bien, esta es
  la Escuela de Ciencias de la Computaci ón, paga por todo esto, verdad?

A veces, la incapacidad para probar o resolver un problema matemático no se debe a la falta de habilidad, sino a la ausencia de una solución debido a la falta de estructura o patrón en el problema. Se introduce el concepto de aleatoriedad lógica, una noción de aleatoriedad basada en la estructura, en lugar de la imprevisibilidad y la aleatoriedad física. Se propone el concepto básico de mirar el tamaño del programa más conciso que calcula algo, medida de su complejidad y contenido de información algorítmica. Se introduce Ω como la probabilidad de que un programa generado al azar se detenga, un número real específico que depende del lenguaje de programación y se considera información matemáticamente irreducible. La idea de Hilbert de formalización completa, a pesar de su fracaso debido a las obras de Gödel y Turing, se ha realizado en los lenguajes de programación de computadoras, omnipresentes en la actualidad.