Decidibilidad

I. Procedimientos Efectivos

La idea de tener una "receta" para resolver una clase de problemas nos relaciona con los procedimientos efectivos.

Las recetas están escritas en algún lenguaje (español, chino, computacional, etc.) alguien o algo debe entenderlas y ejecutarlas; no requiere gran ingenio para su ejecución; terminan en un tiempo finito; dan una respuesta o resultado correcto.

Definición$_1$. Una *Instrucción* es una explicación clara de cómo hacer o ejecutar algo, sin necesidad de ingenio, ni originalidad y son de longitud finita.

Definición$_2$. Un *Procedimiento Efectivo,* PE, es un conjunto **finito** y **ordenado** de instrucciones tales que:

i). No hay elementos al azar en su ejecución.

ii). Algunas veces termina en un número finito de pasos (y por tanto en un tiempo finito).

iii). Cuando termina, da una respuesta correcta.

Debe de sobre-entenderse que:

- El PE debe estar sintáctica, semántica y lógicamente correcto.
- Para lo que se podría necesitar ingenio es para el diseño de un PE.
- Si el PE falla, se debe al (mal) diseño.

Notación:

1. Un procedimiento efectivo (PE) lo denotaremos por

$$PE = \langle I_1, I_2, \dots, I_n \rangle$$

donde $I_1, I_2, \dots, I_n$ son las instrucciones del procedimiento efectivo PE.

2. Cuando aplicamos un PE a un dato inicial, digamos x, lo expresaremos como $PE(x)$. A x se le llama *Entrada.* Quedando claro que $PE(x)$ no es un una función.

3. Algunas veces la respuesta (correcta) a $PE(x)$ es *sí* o *nó.* En tales casos, escribiremos $PE(x) =$ SI o $PE(x) =$ NO.

Observaciones:

i). $\left|\left\{ I \;/\; I \text{ es una instrucción} \right\}\right| =$

$= \left|\left\{ s \;/\; s \text{ es una sucesión finita de símbolos} \right\}\right| \le \aleph_0.$

ii). $\left|\left\{ PE \;/\; PE \text{ es un procedimiento efectivo} \right\}\right| =$

$= \left|\left\{ t \;/\; t \text{ es una sucesión finita de instrucciones} \right\}\right| \le \aleph_0.$

II. Algoritmos

Los algoritmos son un caso particular de procedimientos efectivos, son cuando terminan en un número finito de pasos.

Definición$_3$. Un *Algoritmo* es un procedimiento efectivo que siempre termina en un **número finito** de pasos es decir,

i). No hay elementos al azar en su ejecución.

ii). Siempre termina en un número finito de pasos (y por tanto en un tiempo finito).

iii). Al terminar la respuesta es correcta.

Notación: Siendo consistente con la notación anterior:

$$\mathcal{A} = \langle I_1, I_2, \ldots, I_n \rangle$$

denotará un algoritmo $\mathcal{A}$, con instrucciones $I_1, I_2, \ldots, I_n$.

Ejemplos:

1. Hay un procedimiento efectivo –de hecho un algoritmo– que, dada cualquier expresión ε, decide si es una fórmula o no lo es.

2. Hay un algoritmo que, dado un conjunto **finito** de fórmulas $\Sigma \subseteq \Phi(\mathbb{B}_\rho)$ y otra fórmula α, diga si $\Sigma \models_T \alpha$ o no.
El procedimiento (efectivo) son las tablas de Verdad.

III. Conjuntos Decidibles

En lo que sigue, sea X un conjunto no vacío, $X \neq \emptyset$, de cardinalidad contable, $|X| \leq \aleph_0$, y sean $A, B \subseteq X$.

Un conjunto, A, será decidible si hay un procedimiento efectivo que termine en un número finito de pasos, es decir un algoritmo, que nos diga qué elementos, $x \in X$, pertenecen al conjunto y cuáles no.

Definición$_4$. Diremos que A es *Decidible* syss hay un procedimiento efectivo $\mathcal{A}$ tal, que para cada $x \in X$, se tiene,

i). $x \in A$ syss $\mathcal{A}(x) =$ SI. Y

ii). $x \notin A$ syss $\mathcal{A}(x) =$ NO.

Osérvese que dicho procedimiento en realidad es un algoritmo. Si en la definición anterior pedimos que el procedimiento fuera un algoritmo, no necesitaríamos exigir la condición **ii**). También suele decirse que dicho procedimiento (efectivo) es *de Decisión.*

Ejemplos:

1. Todo conjunto finito es decidible.
Las instrucciones pueden hacer una lista de los elementos (son un número finito) del conjunto; entonces el algoritmo puede cotejar la entrada con la lista.

2. Dado un conjunto **finito** de fórmulas $\Sigma \subseteq \Phi(\mathbb{B}_\rho)$, el conjunto de consecuencias tautológicas de Σ, $\Sigma^{\models_T}$, es decidible. En particular, con $\Sigma = \emptyset$, el conjunto de tautologías, $\mathcal{T}_{\mathbb{B}_\rho}$, es decidible.

3. El $\emptyset$ y el total, X, son decidibles.

4. Si A y B son decidibles, entonces $A \cup B$, $A \cap B$ y $A^c (= X \setminus A)$ son decidibles.

5. Sea $\mathcal{D}(X) = \left\{ D \subseteq X \ / \ D \text{ es decidible} \right\}$. Así.

a. $\emptyset \neq \mathcal{D}(X) \subseteq \wp(X)$.

b. $\langle \mathcal{D}(X), \cup , \cap , X \setminus \_ , \emptyset , X \rangle$ es un álgebra boolena; de hecho una subálgebra del álgebra boolena $\wp(X)$.

c. $\left| \mathcal{D}(X) \right| \leq \aleph_0$. De hecho se tiene,

i. Si $\left| \mathcal{D}(X) \right| = \aleph_0$, el álgebra $\mathcal{D}(X)$, no es más que (salvo isomorfismo) la de Lindenbaum.

ii. Si $\left| \mathcal{D}(X) \right| < \aleph_0$, el álgebra $\mathcal{D}(X) = \wp(X)$.

6. En los números reales $\mathbb{R}$. Dados $a, b, c \in \mathbb{R}$, el conjunto

$$A = \left\{ x \in \mathbb{R} \ / \ ax^2 + bx + c = 0 \right\}$$

es decidible.

IV. Conjuntos Efectivamente Numerables.

El caso más interesante –por cierto– es cuando $|X| = \aleph_0$. Aquí el número de conjuntos **no** decidibles es incontable ($2^{\aleph_0}$) mayor pues que los decidibles. Algunos conjuntos infinitos (numerables) serán decidibles y ¿de los otros? Se puede decir algo más débil.

Definición$_5$. Un conjunto $A \subseteq X$ es *Efectivamente Numerable,* EN syss existe un procedimiento efectivo que produzca una lista, en algún orden, de todos los elementos de A.

Debe de quedar claro que: Si A es infinito, entonces tal vez el procedimiento nunca termine; pero a la larga (es decir, despues de un tiempo finito), cualquier elemento particular de A debe aparecer en la lista.

Definición$_6$. Un conjunto $A \subseteq X$ es *Semidecidible* syss existe un procedimiento efectivo PE tal, que para todo $x \in X$,

$$x \in A \text{ syss } PE(x) = \text{SI}$$

A tal procedimiento se le llama *Semialgoritmo* o *de Semidecisión.*

Si fuera el caso que $x \notin A$, el procedimiento podría producir la respuesta “**NO**”; lo más probable es que continue indefinidamente sin dar una respuesta, pero no debe mentirnos y producir la respuesta “**SI**”.

Proposición. Sea $A \subseteq X$. A es efectivamente numerable syss A es semidecidible.
Prueba:
Si A es efectivamente numerable entonces dado cualquier $x \in X$, podemos examinar la lista de los elementos de A mientras el procedimiento la produce. Solo cuando x aparezca, decimos “SI”. (Obsérvese que si $x \notin A$, no se dá ninguna respuesta -esto es lo que impide que A sea decidible. También obsérvese que si x no ha aparecido en algún momento dentro de los elementos numerados de A, en general no hay manera de saber, más adelante, si x aparecerá o si $x \notin A$.)
A la inversa, supongamos que tenemos un semialgoritmo para A; queremos obtener una lista de los elementos de A. Enumeremos todos los elementos de X, por ejemplo:

$$x_0, x_1, x_2, \ldots$$

y seguimos con el siguiente plan inicial:

1. Dedicamos un minuto a verificar si x_0 es un elemento de A (usando el procedimiento dado).

2. Dedicamos dos minutos a verificar x_0 y luego dos minutos a x_1.

3. Dedicamos tres minutos a verificar x_0, luego tres minutos a x_1 y luego

tres minutos a x_2.
Y así sucesivamente. Cada vez que nuestro procedimiento produzca un "SI", colocamos el elemento de X aceptado en la lista de salida.

De esta manera llegará un momento en que todo elemento de A aparecerá en la lista y en caso de que un elemento de X no esté en A, nuestro plan funciona porque el procedimiento dado, solamente se entretiene en él "unos" minutos. Sin embargo, este plan hay que refinarlo pues, en la lista se repiten -de hecho una infinidad de veces- los elementos de A. †

Algunas observaciones:

1. Todo conjunto decidible es también semidecidible y por tanto también es efectivamente enumerable EN.

2. $\emptyset$ y X son EN.

3. (Teorema de Kleene). A es decidible syss tanto A como su complemento, $X \setminus A$, son EN's.

4. Si tanto A como B son EN's, entonces $A \cup B$ y $A \cap B$ también son EN's.

5. Sea $\mathcal{EN}(X) = \left\{A \subseteq X \ / \ A \text{ es } EN\right\}$. Así,

a. $\mathcal{EN}(X) \neq \emptyset$ y $\mathcal{EN}(X) \subseteq \wp(X)$.

b. $\mathcal{D}(X) \subseteq \mathcal{EN}(X)$.

c. $\left| \mathcal{EN}(X) \right| \leq \aleph_0$.

d. Si $\left| \mathcal{EN}(X) \right| < \aleph_0$, entonces $\mathcal{EN}(X) = \wp(X) = \mathcal{D}(X)$.

Proposición. Sea $\Sigma \subseteq \Phi(\mathbb{B}_\rho)$. Si Σ es un conjunto decidible, entonces el conjunto de sus consecuencias tautológicas, $\Sigma^{\models_T}$, es efectivamente enumerable; en símbolos:

$$\Sigma \in \mathcal{D}(EXP_\rho) \Rightarrow \Sigma^{\models_T} \in \mathcal{EN}(EXP_\rho)$$

Prueba: Basta suponer que Σ es efectivamente enumerable. Consideremos una enumeración de Σ, digamos $\left\{\sigma_i \ / \ i \in \mathbb{N}\right\} = \Sigma$. Ahora bien, dada una fórmula $\alpha \in \Phi(\mathbb{B}_\rho)$, podemos verificar sucesivamente (por medio de las tablas de verdad) si es que:

$$\begin{aligned} \emptyset &\models \alpha, \\ \{\sigma_0\} &\models \alpha, \\ \{\sigma_0, \sigma_1\} &\models \alpha, \\ \{\sigma_0, \sigma_1, \sigma_2\} &\models \alpha, \end{aligned}$$

etcétera. Si alguna de estas condiciones se cumple, entonces contestamos "SI". Si no se cumple entonces continuamos. Este procedimiento es un semialgoritmo, pues en caso de responder "SI" tendríamos el caso digamos $\{\sigma_0, \sigma_1, \ldots, \sigma_n\} \models \alpha$ con

$\{\sigma_0, \sigma_1, \ldots, \sigma_n\} \subseteq \Sigma,$ por lo que $\Sigma \models \alpha.$ Y si $\Sigma \models \alpha,$ entonces -por el corolario al MT de compacidad- hay un subconjunto finito de Σ y tarde que temprano producirá una respuesta afirmativa. †

V. Indecidibles.

Sea X un conjunto numerable, $|X| = \aleph_0$. Así, $|\wp(X)| = 2^{\aleph_0}$.

1. Hay conjuntos indecidibles: $|\mathcal{D}(X)| = |\mathcal{EN}(X)| = \aleph_0$ y $|X \setminus \mathcal{EN}(X)| = 2^{\aleph_0}$.

2. Costruyamos un indecidible: Sean $\{x_i \ / \ i \in \mathbb{N}\} = X$ y $\{A_j \ / \ j \in \mathbb{N}\} = \mathcal{D}(X)$. Dando un $n \in \mathbb{N}$, puede ocurrir que $x_n \in A_n$ o que $x_n \notin A_n$. Definimos:

$$C = \{x_n \in X \ / \ x_n \notin A_n\}$$

(¡La diagonal de Cantor!). Se tiene que:

a. $C \in \wp(X) \setminus \mathcal{D}(X)$ y $|C| = \aleph_0$.

b. $C^c = (X \setminus C) \in \mathcal{EN}(X) \setminus \mathcal{D}(X)$ y $|C^c| = \aleph_0$.

3. $\mathcal{D}(X) \subsetneq \mathcal{EN}(X) \subsetneq \wp(X)$

4. $|\mathcal{D}(X)^c| = |\mathcal{EN}(X)^c| = |\wp(X)| = 2^{\aleph_0}$.

VI. Funciones Calculables.

Sean X, A conjuntos no vacíos, $|X| \leq \aleph_0$ y $A \subseteq X$. Definimos:

$$C_A : X \longrightarrow \{0,1\}$$

$$C_A(x) = \begin{cases} 1 & \text{si} \quad x \in A \\ \text{o} & \\ 0 & \text{si} \quad x \notin A \end{cases}$$

Queremos que:

1. $A \in \mathcal{D}(X)$ syss C_A es *Calculable.*

2. $A \in \mathcal{EN}(X)$ syss C_A es *Parcialmente Calculable.*

¿Cómo definir que una función $f : X \to X$, sea Parcialmente Calculable y cómo, cuándo sea Calculable?