Antonio David Villegas Yeguas - Entrega relación 2

$\boxed{1}$ Sabemos que L es r.e, pero no recursivo, luego $\bar{L}$ no es r.e

Supongamos que L' es r.e.

Podríamos construir una MT $M_{\bar{L}}$ para $\bar{L}$ tal que una entrada w, añade un 1 al principio y simule L' sobre $1w$. Si la simulación acepta, estaríamos aceptando $\bar{L}$, luego esto se contradice con que $\bar{L}$ no es r.e, luego L' (o su complementario) no puede ser r.e.

$\boxed{2}$

a) Este problema podemos comprobar que es semidecidible, ya que podemos construir una MT que resuelva el problema (aunque no es decidible ya que el ~~c~~ conjunto a comprobar no es finito, al poder probar con infinitas palabras). La MT M' sería la siguiente: (M es la máquina dada):

Escoger de forma no determinista una palabra w_0:

Si M acepta w_0 (Usando ~~una~~ la máquina universal de Turing para simularlo):

Escoger de forma no determinista una palabra w_1 tq $w_0 \neq w_1$

Si M acepta w_1:

Aceptar

Si no:

Rechazar

Si no:

Rechazar.

Por lo que este problema es semidecidible. Como hemos comentado, no es decidible, y para demostrarlo, al ser semidecidible (r.e), podemos usar el Teorema de Rice, ya que el aceptar dos palabras diferentes es una propiedad no trivial del lenguaje ya que aunque tengamos dos máquinas de Turing distintas, ambas deben aceptar las mismas palabras si trabajan sobre el mismo lenguaje.

b) Este problema es no semidecidible, y para demostrarlo voy a hacer una reducción del problema C-Parada, el cual sabemos que es no semidecidible:

C-Parada: Dada una MT M y una entrada W, determinar si M cicla con entrada W.

Finito: Dada una MT M'. acepta un número finito de palabras distintas.

Una instancia de C-Parada es (M, w) el objetivo es construir una MT F(M,W) tal que reduzcamos el problema de si un lenguaje es finito (es regular) al problema de C-Parada:

Máquina F(M, w): La entrada está en la 1ª cinta, M en la 2ª y w en la 3ª
Entrada $v \in \{0,1\}^*$
Calculamos |v| y la escribimos en la 4ª cinta.
~~Simulamos~~
Escribimos en la 5ª cinta M111M
Simulamos |v| pasos de M111M con la máquina universal de turing.
Si M111 para tras |v| pasos:
Aceptar V
Si no
Rechazar V

Con esta MT F(M,w) vemos que M representada como palabra ∈ C-Parada si y solo si $L(F(M,w)) = \emptyset$ si y solo si $L(F(M,w)) \in$ lenguajes regulares (es finito)
Luego en los casos negativos de C-Parada (se detiene la máquina), se acepta V, luego $L(F(M,w)) \neq \emptyset$, luego se rechaza.

En los casos positivos de C-Parada (no se detiene), se rechaza V, luego $L(F(M,w)) = \emptyset$, luego se acepta.

Como sabemos que C-Parada es no semidecidible y el problema dado se reduce a resolver C-Parada, el problema dado también es no semidecidible.

c) Este problema es no semidecidible, y para demostrarlo voy a hacer una reducción desde C-Universal, ya que como hemos visto, Universal es semidecidible pero no decidible, por lo que C-Universal es no semidecidible.

C-Universal: Dada una MT M y una palabra w, determinar si M no acepta w.

Independiente-contexto: Dada una MT M', determinar si el lenguaje aceptado es independiente del contexto.

Para esto construiremos la siguiente MT F(M, w), que dada una instancia del problema universal (M, w), F(M, w) ~~es~~ pasa a ser una instancia de nuestro problema.

Máquina de Turing F(M, w)
Entrada: palabra $v \in \{0,1\}^*$
    Calculamos $|v|$ y lo almacenamos en la 2ª cinta
    Borramos v de la primera cinta y ponemos M 111 w
    Simulamos $|v|$ pasos de M con entrada w
    Si M acepta w en $|v|$ pasos:
        Aceptar v
    Si no:
        Rechazar v

Con esto vemos que:

- En los casos negativos de C-Universal (M acepta w), aceptamos cualquier palabra, donde hay palabras no independientes del contexto, luego también es un caso negativo de nuestro problema.

- En los casos positivos de C-Universal (M no acepta w), no aceptamos nada y el lenguaje es vacío: $L(F(M,w)) = \emptyset$, y como sabemos, el lenguaje vacío es independiente del contexto, luego es un caso positivo de nuestro problema.

Vemos como resolver el problema se reduce a resolver C-Universal, y como C-Universal es no semidecidible, este problema también es no semidecidible.

d) En este caso el problema es decidible, ya que en un número finito de pasos somos capaces de aceptar o rechazar, y nunca vamos a ciclar.

Para los casos en los que la máquina para, o supera las 10 casillas de la cinta es facil de comprobar, el problema lo tenemos en los casos en los que cicla usando 10 casillas o menos.

Si cicla en menos de 10 casillas tambien lo podemos saber, ya que el número de ~~cel~~ transiciones antes de que cicle es finito.

Teniendo en cuenta una única casilla, el número de transiciones que puede aplicar es el numero de estados por el numero de simbolos del alfabeto de trabajo. Como son 10 casillas, el número de transiciones será $10 \cdot$ numero de simbolos^{10} $\cdot$ número de estados, el primer 10 ~~por el~~ sería el numero de casillas a comprobar, numero de simbolos^{10} ya que en total, podemos ~~pasar por cada una de~~ hacer ese numero de combinaciones de simbolos, y por último, el número de estados, ya que podemos pasar de un estado a otro sin modificar los simbolos y no estariamos ciclando. Dicho esto, podriamos diseñar el siguiente algoritmo que resuelve el problema:

Maquina de Turing M:
Entrada: 0011
  Escribimos en la 2ª cinta M1110011
  Escribimos en la 3ª cinta un 0
  Mientras que la 3ª cinta sea menos que $10 \cdot n^{o}\,simbolos^{10} \cdot n^{o}\,estados$
    Simulamos tantos pasos de M111 0011 como diga la 3ª cinta
    Si supera en 10 el número de casillas usadas en la simulacion
      Rechazamos
    Si para y usa menos de 10 casillas
      ~~Incrementamos en 1 el valor de la 3ª cinta~~
      Aceptamos

    Si no para
      Incrementar en 1 el valor de la 3ª cinta

  Aceptamos (Ha hecho todas las transicion posibles y no usa más de 10, esta ciclando)

e) Para resolver este problema tenemos que comprobar q dadas dos MT M_1 y M_2 $L(M_1) \subseteq L(M_2)$ y que $L(M_2) \subseteq L(M_1)$.

Como hemos visto en el manual dado sobre decibilidad y semidecibilidad, el problema de determinar si $L(M_1) \subseteq L(M_2)$ es no semidecidible, este problema, como se ~~reduce~~ trata de resolver dos veces un problema no semidecidible, tampoco es semidecidible.

f) Este problema es decidible, ya que podemos diseñar un algoritmo que en un número finito de pasos nos de una respuesta:

Maquina de Turing M:
Entrada: Palabra u

Escribir $|u|$ en la 2ª cinta.
Escribir 0 en la 3ª cinta
Mientras el valor de la 3ª cinta $< |u|$
Simular M111 u durante el número de pasos que indique la 3ª cinta
Si acepta:
Aceptar
Si no
~~Rechazar~~
Incrementar en 1 el valor de la 3ª cinta

Rechazar (Hemos simulado $|u|$ pasos y todavía no ha aceptado, no se cumple)