Correcciones del Documento anterior y formalizaciones.

**Resumen**

Aquí vamos a formalizar las ideas que hemos expuesto en el documento padre de este mismo, y vamos a dar los teoremas que demuestran la corrección del algoritmo que estas ideas construyen.

En esta parte del Documento, vamos abordar el problema formalizando las nociones que van a hacer que tengamos una subrutina, que es el que va a dar el control en la estructura recursiva, entonces definamos dichas nociones de la siguiente manera: Sea una maquina de Turing $MT$ y una cadena que es aceptada por dicha maquina de Turing, con los atributos que tiene la maquina de Turing, entonces definimos la noción de localidad en $DC_{LOCAL}$ que denotaremos como simplemente como $LOCAL$.

**Definiciones**

**Definition 0.1.** *Sea $v \in VG$ un proceso(nodo) en el modelo Distribuido, vamos a denotar lo siguiente:*

$$(w_i, v) \tag{1}$$

*esto va a denotar que el elemento de la cadena de $w$ tiene localidad $v$, o en otros términos que a $v$ le pertenece la cadena de localidad $i$*

**Remark.** *Observemos que de la definición anterior se desprende una relación entre los elementos de $\psi$ tal y como lo observamos en la siguiente definición*

**Definition 0.2.** *Sean $v \in VG$ y $w_i \in w$ entonces decimos que están relacionados si y solo si $(w_i, v)$*

**Remark.** *En esta parte podemos construir un conjunto, que es el que esta formado por los elementos de la cadena que tienen localidad $p \in V(G)$ de la siguiente manera: $loc(P) = \{wi : (w_i, P)\}$ Este conjunto esta encapsulando las nociones anteriormente Descritas.*

Entonces una vez que tenemos esto bien definido, podemos hacer lo siguiente:
De manera local sea $(w_i, v)$ entonces $v(w_i)$ que es la asignación que tiene sentido, ya que estamos en un ambiente distribuido, y asi es la naturaleza de la distribución de la información.

**Descripción del diseño del algoritmo**

Una vez que tenemos la pareja entonces tenemos de manera computacional y localmente lo siguiente $v(w_i)$ que es el procedimiento computacional(Distribuido) que hemos de desarrollar tal que solucione nuestro problema con la entrada de localidad asignada, entonces lo que haremos primero es construir un loop que haga una llamada recursiva de $\delta$ hasta que la cadena que nos regrese dicho proceso recursivo, sea distinta a la de la localidad actual. Pero para ello construiremos una subrutina booleana que lo que hará es un token en las entradas via nuestra noción de localidad.

Una vez que tenemos esta subrutina de tipo booleana diseñamos el algoritmo que nos demostrara que $w$ es una cadena aceptada por el modelo **LOCAL**, y esto a un alto nivel demostrara un lado de nuestro teorema principal.

Asi que este es la simulación distribuida de la maquina de Turing. Una vez que tenemos el diseño de la simulación, es entonces hacer un test de corrección.

**Algorithm 1** $func\_boolean(w_i, p)$

**if** $(w_i, p)$ **then**
  **return** $true$
**else**
  **return** $false$
**end if**

**Algorithm 2** $Simulate\_Algo\_TM(w, G, TM)$

$w \leftarrow (w_1, \ldots, w_n)$
**Síncronamente**
**for all** $r = 1$ **to** to $n$ **do**
  $p(w_i)$
  $q \leftarrow q_0$
  **while** $func - boolean(w_i, p)$ **do**
    **call** $\delta(q, w_i)$
    $(q_r, w_r, P) \leftarrow \delta(q, w_i)$
  **end while**
  **if** $q_r = q_{accept}$ **then**
    **return** $q_r$
  **else**
    $M \leftarrow < q_r, w_r, P >$
    **send** $M$
  **end if**
**end for**

**Theorem 0.1.** *El algoritmo* $Simulate\_Algo\_TM()$ *es correcto*

*Demostración.* Sea $i = r$ una roda en la ejecución del algoritmo.
En esa ronda tenemos la siguiente estructura de dato $M =< w_r, q_r, P >$ donde
$P \in \{LEFT, RIGHT, STAY\}$.

Para esa ronda tenemos que el $w_r$ que es el resultado de la llamada recursiva de $\delta$, podemos concluir que $func - boolean(w_r, p) = false$ por lo tanto no es de la localidad de $p$, entonces podemos tomar las siguientes decisiones:
Como $w_r \in w$ es decidible en el sentido de la maquina de Turing, entonces $q_r$ puede que sea el estado de aceptación, en ese caso el algoritmo toma la decision $return\ q_r$, en su defecto toma la decision de enviar el mensaje $M$ a sus vecinos. para el cual como $TM$ es tal que $w$ es decidible entonces en la ronda $r = i + 1$ se tendrá el estado de aceptación para un $p$ proceso de $LOCAL$.
Por lo tanto existe un proceso $p$ tal que regresa un estado de aceptación.
Por lo tanto $LOCAL$ es tal que hace a $w$ decidible en el sentido Distribuido.
Por lo tanto $Simulate\_algo\_TM$ es correcto. □