Tarea 3 - Cómputo distribuido

Dalia Camacho, Gabriela Vargas

1 Topología de un problema en los primeros tres estados

Consideramos el caso en que Bob y Alice quieren reunirse. Cada uno puede preferir verse a las $9a.m.$ o a las $9p.m.$, en el primer estado únicamente conocen su preferencia. Esto se puede ver en la gráfica G_0.

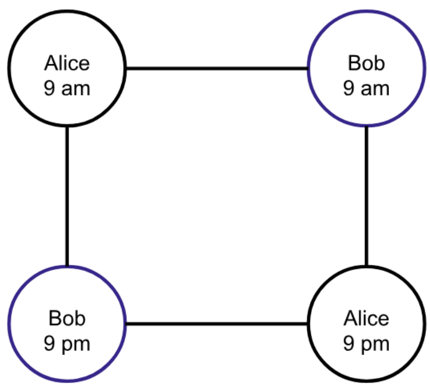

Figure 1: Gráfica G_0 que representa el conocimiento de Bob y Alice al tiempo 0, cuando ambos pueden tener dos opciones.

En el segundo estado Bob puede conocer la preferencia de Alice o si el mensaje no llega, únicamente su preferencia. Como Alice envía primero el mensaje ella sólo sabe lo que ella quiere. Esto se puede ver en la gráfica G_1.

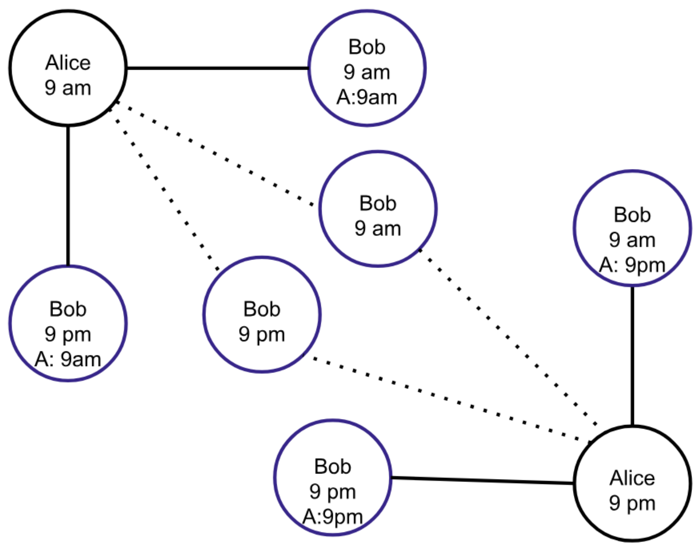

Figure 2: Gráfica G_1 que representa el conocimiento de Bob y Alice al tiempo 1, cuando ambos pueden tener dos opciones y Alice envía primero el mensaje.

En el tercer estado ambos pueden conocer la preferencia del otro si a Bob le llega el mensaje de Alice y a Alice el mensaje de Bob; pueden sólo conocer lo que ellos quieren si el primer mensaje de Alice falló; o bien que sólo Bob conozca lo que ambos quieren por que la respuesta que le envía a Alice falla. Esto se puede ver en la gráfica G_2.

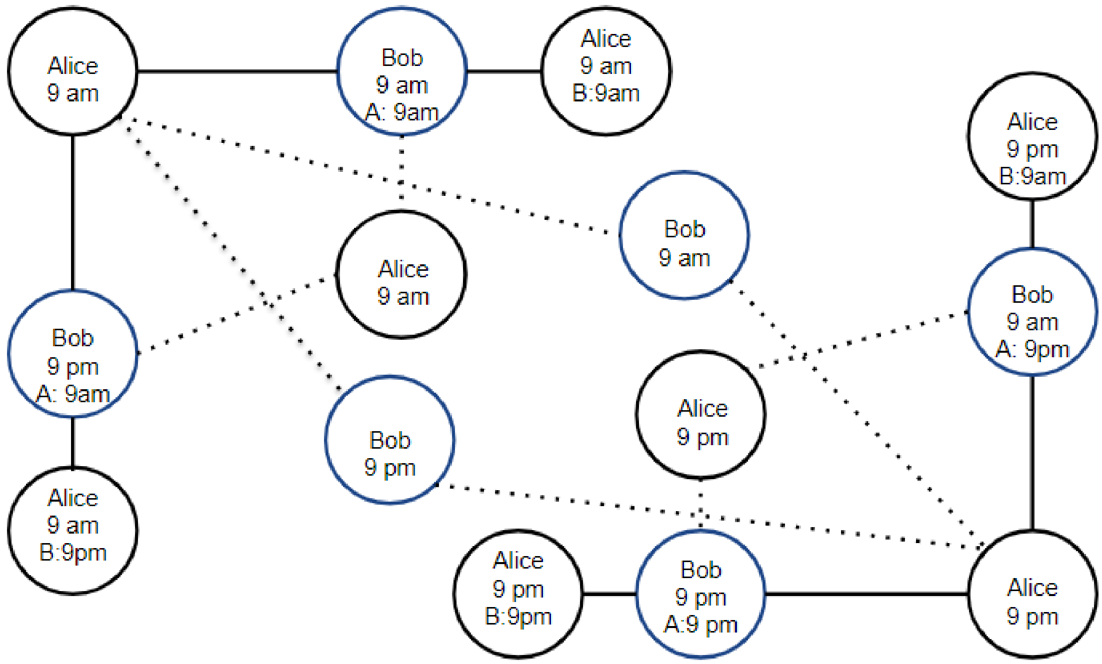

Figure 3: Gráfica G_2 que representa el conocimiento de las preferencias de la otra persona en el caso de que los mensajes lleguen con éxito o de que se presente una falla en la comunicación

2 Problema de consenso

2.1

Considere el siguiente algoritmo para el problema de consenso:

1. Define X_i as an input such that $X_i \in X$
2. Send X_i to all $P_1, P_2, ..., P_n$
3. Wait until at least $n - t$ messages have been received.
4. Let $V[j]$ be the value received from process j
5. Return $h(v)$ = maximum value in V = d_i

2.2 Defina el problema de consenso en términos de Agreement, Validity y Termination

- **Agreement:** Indica que las decisiones correctas son las mismas, es decir, $\forall$ P_i y P_j que lleguen a la línea 5 , $di = dj$.

- **Validity:** La decisión es un insumo de cada proceso, es decir, si $d_i \neq \emptyset \Rightarrow \exists\ d_i = X_j$ para algún $X_j \in X$.

- **Termination:** Todos los procesos deciden correctamente, es decir, todos los procesos que no fallan llegan a la línea 5 y entregan un valor d_i.

2.3 ¿En qué casos no sirve el algoritmo?

- Hay más de t fallas.

- Más de t participantes no envíen el mensaje.

- Si uno de los participantes no recibe el valor máximo en los primeros $n-t$ mensajes, éste participante no elegirá el máximo como valor de salida y no se logrará consenso.

- Si el mensaje que contiene el valor máximo no le llega a los demás participantes se podría llegar a un consenso, pero sin haber elegido el verdadero valor.

2.4 ¿Para qué condiciones de entrada C el algoritmo funciona?

- Funciona cuando todas las entradas son iguales, es decir $X_i = X_j, \forall\ i, j$, por lo que el conjunto C_{equal} está definido de la siguiente forma $V \in C_{equal} \Leftrightarrow V[i] = V[j], \forall\ i, j$.

- Cuando al menos hay $t+1$ X's que comparten el valor del máximo. Por lo que el conjunto C_{in} que contiene todas las posibles entradas V está definido como $C_{in} = \{$Todos los vectores V de n entradas sobre X donde $y = \max(V) \Rightarrow y$ aparece al menos $t+1$ veces.$\}$

3 Autómata determinista finito

Crea dos autómatas deterministas finitos que acepten el lenguaje de las que terminen en 011 constriudas con el alfabeto $\Sigma = \{0, 1\}$. El primero sólo tiene la condición de aceptar el lenguaje y el segundo debe ser el de tamaño mínimo

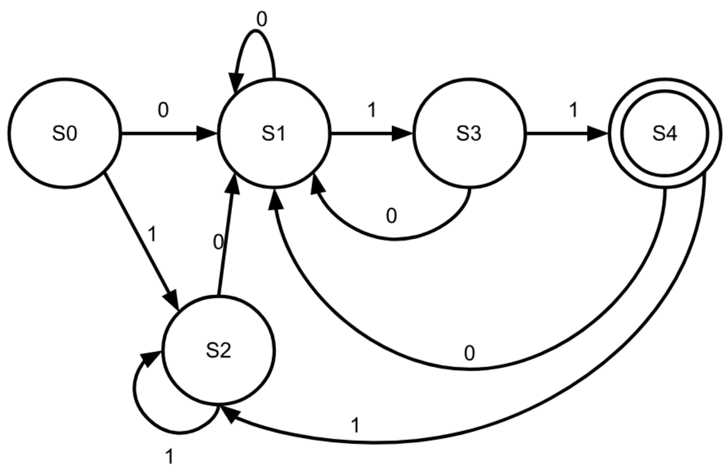

Figure 4: Autómata determinista finito que acepta el lenguaje cuyas cadenas terminan en 011

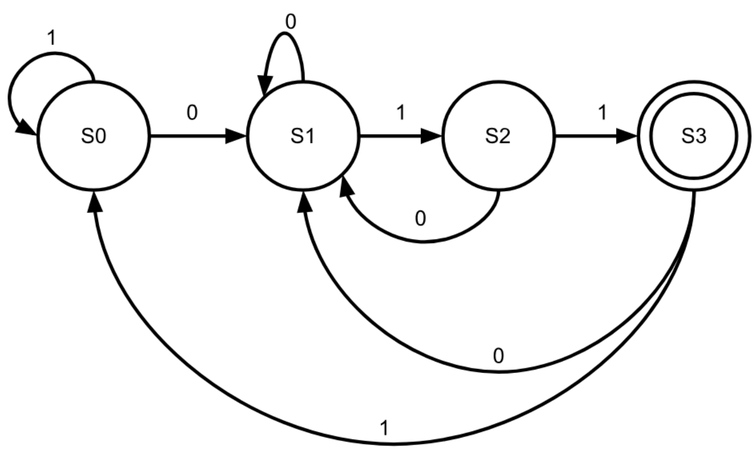

Figure 5: Autómata determinista finito de tamaño mínimo que acepta el lenguaje cuyas cadenas terminan en 011

4 Resumen de la clase

En esta clase revisamos los conceptos y la notación para describir un problema de consenso bajo un modelo de tiempo asíncrono:

- n: número de procesos
- t: número de procesos que pueden fallar *(crash)*

- X^n se refiere a todos los posibles vectores de entrada para los n procesos que integran la red.
- X denota el conjunto de valores posibles de entrada.
- $P = \{P_1, P_2, ..., P_n\}$ hace referencia al conjunto de procesos pertenecientes a la red, donde cada P_i representa un proceso.

Alineado con esta notación, el problema de consenso quedaría descrito de la siguiente forma:

1. Define X_i as an input such that $X_i \in X$
2. Send X_i to all $P_1, P_2, ..., P_n$
3. Wait until at least $n - t$ messages have been received.
4. Let $V[j]$ be the value received from process j
5. Return $h(v)$ = maximum value in V = d_i

En la segunda parte de la clase, empezamos a ver el concepto de *indistinguibilidad*, que se define como una falta de conocimiento sobre el funcionamiento interno de los componentes de un sistema. Este fenómeno ocurre como consecuencia de la existencia de ambientes locales perteneciente a un ambiente global.

En teoría de autómatas, la indistinguibilidad se define en el teorema de Moore de la siguiente forma: "Dada una máquina S y una serie de experimentos realizados en S, existe otra máquina experimentalmente distinguible S para la cuál el experimento original producirá la misma salida."

References