Tarea 6 - Cómputo distribuido

Dalia Camacho, Gabriela Vargas

1 Escribir y probar el algoritmo de acuerdo aproximado para $\frac{1}{2^k}$ para A y B con el modelo de comunicación R_1 para k rondas

Para resolver el problema de consenso mediante un acuerdo aproximado con k para el modelo de comunicación R_1 se propone el siguiente algoritmo para cada proceso P_i.

$v_i \leftarrow$ Valor inicial de P_i
for $j \leftarrow 1$**to** k **do**
  El proceso v_i envía P_i
  if *Recibes* v_m *del otro proceso* **then**
    $v_i \leftarrow \frac{v_i+v_m}{2}$
  end
  El proceso P_i toma la decisión $dec_i = v_i$.
end

Algorithm 1: Algoritmo para llegar a un acuerdo aproximado para k entre dos procesos.

Ahora probamos que el algoritmo cumple con validez, acuerdo y terminación. Para el acuerdo aproximado con k tenemos lo siguiente:

- **Validez**: Si ambos procesos inician con el mismo valor, entonces deciden por su valor inicial.
- **Acuerdo**: Los procesos A y B llegan a un acuerdo si $|dec_A - dec_B| \leq \frac{1}{2^k}$
- **Terminación**: Ambos procesos toman una decisión.

El algoritmo cumple con validez, ya que si $v_A = v_B$ y uno de los procesos recibe el valor del otro entonces $\frac{v_A+v_B}{2} = \frac{v_A+v_A}{2} = v_A = v_B$, en otro caso simplente deciden por v_A o por v_B.

Se cumple con terminación ya que al final de k rondas ambos procesos toman una decisión.

Ahora demostramos que bajo este algoritmo de consenso $|dec_A - dec_B| \leq \frac{1}{2^k}$ se cumple en k rondas. Esto lo demostraremos utilizando el principio de inducción.

1. Primero demostramos que el algoritmo funciona para $k = 0$ y para $k = 1$. Para $k = 0$ los procesos A y B eligen su valor inicial, ya sea 0 o 1 y $|dec_A - dec_B| \leq |1-0| = 1 = \frac{1}{2^0}$, por lo que se cumple. Ahora consideremos el caso donde $k = 1$, en la Figura 1 mostramos el esquema de comunicación, el valor inicial y la decisión que toma cada nodo.

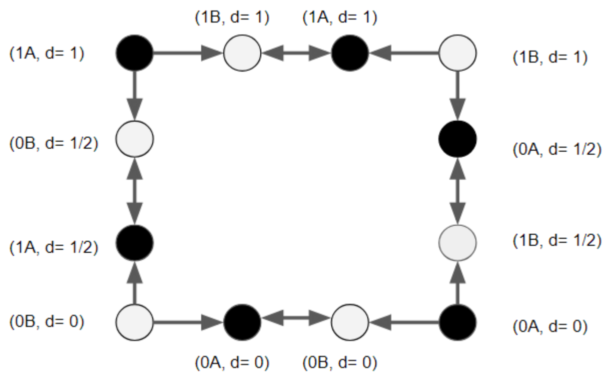

Figure 1: Acuerdo aproximado para $k = 1$

Podemos notar que la decisión que toman cualesquiera dos nodos unidos por una arista difieren en su decisión en a lo más $\frac{1}{2} = \frac{1}{2^1}$. Por lo que el algoritmo llega a un 1-acuerdo para $k = 1$.

2. Ahora suponemos que el algoritmo funciona para $k - 1$, donde $k - 1 \geq 1$. Si lo anterior se cumple entonces la distancia entre cualesquiera dos nodos en la ronda $k - 1$ es a lo más $\frac{1}{2^{k-1}}$.

3. Ahora probamos que el algoritmo funciona para k rondas. Dado lo anterior en la ronda $k-1$, $|v_A - v_B| \leq \frac{1}{2^k-1}$, sin perdida de generalidad supongamos que en el peor de los casos $v_A = x$ y $v_B = x + \frac{1}{2^{k-1}}$ para dos nodos unidos mediante una arista. Si nos enfocamos en ese caso la comunicación entre estos nodos en la ronda k se puede representar como se muestra en la Figura 2.

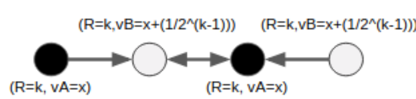

Figure 2: Comunicación en la ronda k para dos nodos cuyo valor se encuentra a una distancia de $\frac{1}{2^{k-1}}$.

Los nodos que no reciben información deciden por su valor inicial, mientras

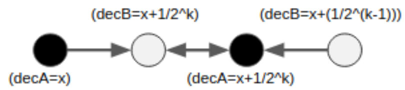

Figure 3: Decisión en la ronda k.

que los que reciben información deciden por el promedio que en este caso seria $\frac{x+x+\frac{1}{2^{k-1}}}{2} = x + \frac{1}{2^k}$. Con esto las decisiones se ven de la siguiente forma.

Podemos ver que o ambos nodos deciden por $x + \frac{1}{2^k}$ o bien A decide por x y el B por $x + \frac{1}{2^k}$ con lo que $|dec_A - dec_B| = |x - (x + \frac{1}{2^k})| = \frac{1}{2^k}$. O bien A decide por $x + \frac{1}{2^k}$ y B por $x + \frac{1}{2^{k-1}}$, con lo que $|dec_A - dec_B| = |(x + \frac{1}{2^k}) - (x + \frac{1}{2^{k-1}})| = |\frac{1}{2^k} - 212^k| = \frac{1}{2^k}$. Por lo tanto se cumple con el acuerdo aproximado para k en k rondas.

Con esto queda demostrado que el algoritmo 1 de acuerdo aproximado resuelve el acuerdo aproximado para k en k rondas.

2 Dibujar el complejo de entrada y salida del problema de consenso aproximado para los nodos {A,B.C} y k=2 en el complejo R1

En este modelo, los nodos A, B, C pueden tomar tres tipos de decisiones {0,1,2}. En R1, el complejo de entrada contiene las posibles preferencias de todos los nodos y se describe de la siguiente manera: decA,decB,decC.

El complejo de entrada está compuesto por las aristas: {0,0,0}, {0,0,1}, {0,0,2}, {0,1,0},{0,1,1}, {0,1,2}, {0,2,0},{0,2,1}, {0,2,2}, {1,0,0}, {1,0,1}, {1,0,2}, {1,1,0},{1,1,1}, {1,1,2}, {1,2,0},{1,2,1}, {1,2,2}, {2,0,0}, {2,0,1}, {2,0,2}, {2,1,0},{2,1,1}, {2,1,2}, {2,2,0},{2,2,1} y {2,2,2}

En tercera dimensión, el complejo se ve de la siguiente manera:

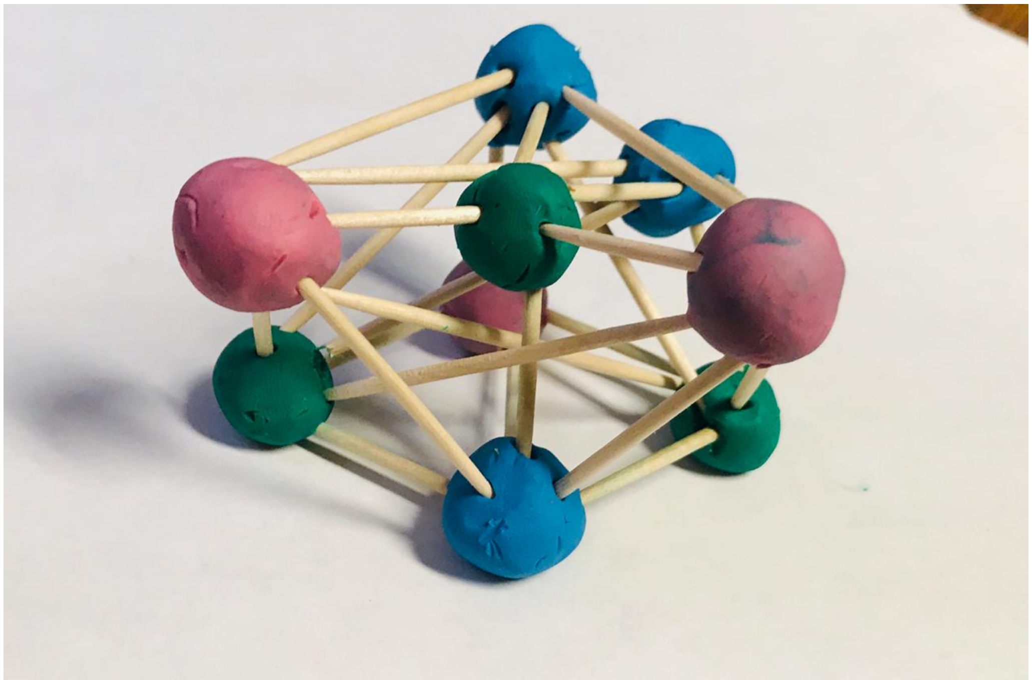

Figure 4: Complejo de entrada.

Por otra parte, el complejo de salidas solo contiene aquellas aristas donde la decisión debe ser la misma al menos para 2 de los 3 nodos. Por lo tanto, las aristas que forman parte de este complejo son:
{0,0,0}, {0,0,1}, {0,0,2},{0,1,0},{0,1,1}, {0,2,0}, {0,2,2}, {1,0,0}, {1,0,1}, {1,1,0},{1,1,1}, {1,1,2},{1,2,1}, {1,2,2}, {2,0,0}, {2,0,2}, {2,1,1}, {2,1,2}, {2,2,0},{2,2,1} y {2,2,2}

3 Demostrar que si hay 2 rondas consecutivas en las que si un proceso p_i escucha del mismo conjunto de procesos, entonces no puede recibir nueva información

Sea UP^r el conjunto de procesos que no han fallado al principio de la ronda r y $R_i(r) \subseteq UP^r$ el conjunto de procesos de los cuales se ha escuchado un mensaje en la ronda r, tenemos que $R_i(0) \supseteq \cdot \supseteq R_i(r-1) \supseteq UP^r \supseteq R_i(r) \supseteq \cdot$. Si, $R_i(r-1) = R_i(r)$ entonces se puede concluir que $R_i(r-1) = UP^r = R_i(r)$. Al final de la ronda r cada proceso sabrá que tiene la información completa porque no ha recibido nueva información, sin embargo, no sabe si los otros p_j saben lo mismo, así que enviará la información que tiene a los demás nodos en la ronda $r+1$, junto con un aviso para ejecutar la toma de decisión.

Lemma: Si un sistema con presencia de falla bizantinas presenta $t < f$ fallas, donde f es el número máximo de fallas toleradas, los procesos terminarán en la ronda $f + 2$.

Demostraremos este Lemma por contradicción. Supongamos que hay un proceso activo que no ha terminado en la ronda $t + 2$. Se sabe que hay un conjunto de procesos activos $\{P_j\}$ donde $1 \leq j \leq t$ tal que, para cada j, todos los procesos activos tienen el valor enviado por p_j en su vector de decisiones. Dado que hay exactamente t fallas, el input de cada nodo fallido será parte del vector de deecisiones de todos los procesos activos al final de la ronda t+1.

4 Resumen de la clase

En esta clase revisamos los resultados fundamentales del algoritmo de consenso con dos procesos bajo los modelos de comunicación vistos en las clases anteriores. Concluimos que bajo el modelo R_1 es imposible resolver el problema de consenso, ya que cualquier canal puede fallar en cualquier ronda y no se puede determinar un algoritmo de decisión que lo resuelva, debido a que la gráfica es siempre conexa sin importar las rondas de comunicación. Esto está determinado por el *FLP Impossibility result.* Sin embargo es posible resolverlo mediante el acuerdo aproximado y definir que la decisión sea arbitrariamente cercana. Vimos también que para n procesos también se puede tomar un $k-$acuerdo en los n procesos a lo más deciden por k valores distintos. El $k - acuerdo$ sólo tiene sentido si $k < n$. Continuamos viendo el consenso en sistemas síncronos, donde el canal de comunicación es confiable, pero en el que los procesos pueden morir. En este tipo de sistemas hay tres posibles tipos de fallas *crash failures* donde muere un proceso. Fallas por omisión donde un proceso no envía o no recibe el mensaje o bien fallas bizantinas donde un proceso envía mensajes arbitrariamente malignos. Para poder hacer frente a fallas por omisión o fallas bizantinas, sólo los procesos correctos deciden.

Volvimos a revisar el ejemplo en que los procesos mandan novedades a todos los demás procesos hasta que todos los procesos tienen la misma información y por lo tanto son capaces de tomar una decisión. Si hay f fallas posibles se necesitan a lo más $f + 1$ rondas para tomar la decisión. Sin embargo se puede terminar antes si todos los procesos reciben en dos ocasiones seguidas información de los mismos nodos. Como demostramos en la pregunta tres de la tarea.

Además vimos el ejemplo del avión que se cayó. Éste tenía un sensor que cuando el avión parecía estar inclinado con la parte delantera hacia arriba, automáticamente redirigía al avión para estabilizarlo. El problema fue que el sensor falló y mandó el avión hacia abajo cuando el avión estaba en una posición correcta. Al ser un solo sensor la probabilidad de fallo era mayor y no se les indicó a los pilotos del sensor y del redireccionamiento automático.