Pontificia Universidad Católica de Chile
Matías Fernández - matias.fernandez@uc.cl

IIC2213 - Lógica para ciencia de la computación

Ayudantía 2 - Viernes 24 de Marzo del 2023

Problema 1. El módulo de conflictos de un sistema operativo funciona de la siguiente forma. Se necesitan correr procesos $p_1, \ldots, p_n$, en T milisegundos (ms); por simplicidad asumimos que cada proceso toma exactamente 1 ms en completarse.

El proceso de planificar los procesos consiste en asignar, a cada proceso, un índice de tiempo entre 0 y $T-1$, que corresponde al tiempo donde se ejecutará el proceso, de forma que alcancen a completarse todos. El problema es que los procesos compiten por recursos de memoria de la CPU. La memoria cuenta con 8 recursos aritméticos distintos: $r_1, \ldots, r_8$, y cada uno de los procesos $p_1, \ldots, p_n$ usa algunos de estos recursos. Esta información viene dada por una función f que asigna a cada proceso un conjunto $f(p_i) \subseteq \{r_1, \ldots, r_8\}$ de recursos.

El problema a resolver es: ¿Hay alguna forma de poder planificar los procesos requeridos en T milisegundos? Tu meta para esta pregunta es mostrar como podemos resolver este problema con un SAT solver: Dados procesos $p_1, \ldots, p_n$, una cantidad T de milisegundos y una función f que asigna a cada proceso un conjunto $f(p_i) \subseteq \{r_1, \ldots, r_8\}$ de recursos, decidir si acaso cxiste una forma de asignar a cada proceso un índice de tiempo entre 0 y $T-1$, de manera que, para cada par de procesos p y p' asignados al mismo tiempo $t(0 \leq t \leq T-1)$, se tiene que $f(p) \cap f(p') = \emptyset$: los procesos no usan ningún recurso en común.

Específicamente, debes mostrar como construir, para inputs $p_1, \ldots, p_n, T$ y f descritos arriba, una fórmula φ tal que φ es satisfacible si y solo si existe una forma de planificar los procesos como se describió más arriba.

Solución: Dados:

- $P = \{p_1, \ldots, p_n\}$ el conjunto de procesos.
- T milisegundos.
- $R = \{r_1, \ldots, r_8\}$ recursos.
- $f(p_i) \subseteq \{r_1, \ldots, r_8\}$ Por construir, fórmula φ satisfacible si y solo si existe una forma de planificar estos procesos.

Para i en $\{1, \ldots, n\}, t$ en $\{1, \ldots, T\}, j$ en $\{1, \ldots, 8\}$, definimos proposiciones a_{it}, b_{itj}y s_{ij}. La intuición es que una valuación de esas variables representa un intento de planificación, de forma que las variables asignadas a 1 tienen la siguiente intuición:

- $a_{it} = 1$ si el proceso p_i es asignado al tiempo $t \in \{0, \ldots, T-1\}, a_{it} = 0$ en otro caso.
- $b_{itj} = 1$ si el proceso p_i es usa el recurso j en el tiempo $t. b_{itj} = 0$ en otro caso.

- $s_{ij} = 1$ si $r_j \in f(p_i)$, $s_{ij} = 0$ en otro caso.

Nuestra fórmula a construir es la únion de las siguientes expresiones: $\varphi_0 = \bigwedge_{r_j \in f(p_i)} s_{ij} \wedge \bigwedge_{r_j \notin f(p_i)} \neg s_{ij}$ (La valuación a las variables s se comporta de acuerdo a los parámetros que nos dieron)

$$\varphi_1 = \bigwedge_{i=1}^{n} \bigvee_{t=0}^{T-1} a_{it} \text{ (todos los procesos son asignados)}$$

$$\varphi_2 = \bigwedge_{i=1}^{n} \left(a_{it} \rightarrow \bigwedge_{t'=0, t \neq t'}^{T-1} \neg a_{it'} \right) \text{ (procesos son asignados a un solo tiempo)}$$

$$\varphi_3 = \bigwedge_{t=0}^{T-1} \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{8} (s_{ij} \wedge a_{it}) \rightarrow b_{itr} \text{ (procesos usan los recursos que les corresponden)}$$

$$\varphi_4 = \bigwedge_{t=0}^{T-1} \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{8} \left(b_{itj} \rightarrow \bigwedge_{i'=1, i' \neq i}^{n} \neg b_{i'tj} \right) \text{ (los procesos no comparten recursos)}$$

Construimos $\varphi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3 \wedge \varphi_4$, por demostrar que φ cs satisfacible si y solo si cxiste una forma de planificar estos procesos. ($\Rightarrow$), por contradicción suponemos φ satisfacible y no se pueden planificar estos procesos de forma que si p y p' son asignados al mismo tiempo entonces $f(p) \cap f(p') = \emptyset$.

De la definición anterior y sin pérdida de generalidad, podemos suponer que p_1 y p_2 son asignados a $t = 0$, pero $f(p) \cap f(p') \neq \emptyset$ (necesariamente se tiene que cumplir para algún tiempo y par de procesos). Nuevamente, sin pérdida de generalidad (pues al menos deben compartir un recurso), supongamos que $r_1 \in f(p)$ y $r_1 \in f(p')$.

Como φ es satisfacible, por construcción sabemos que $a_{10} = 1, a_{20} = 0$ (por φ_1, φ_2), $b_{101} = 1$ y $b_{201} = 1$ (por φ_3). Pero si $b_{201} = 1$ entonces, por φ_4, necesariamente $b_{101} = 0 \rightarrow\leftarrow$. (1)

($\Leftarrow$), por demostración directa, suponemos que existe una forma de asignar a todos los procesos de manera que si p y p' son asignados al mismo tiempo cntonces $f(p) \cap f(p') = \emptyset$. φ_0 es satisfacible asignando $s_{ij} = 1$ si $r_j \in f(p_i)$, $s_{ij} = 0$ en otro caso. Como asignamos cada proceso a solo un tiempo, φ_1 y φ_2 se cumplen asignando a_{it} para cada proceso y tiempo trivialmente. Además, podemos asignar $b_{itj} = 1$ para cada $a_{it} = 1$ tal que $r_j \in f(p_i)$ y $b_{itj} = 0$ en los otros casos. De esta manera, si repetimos lo anterior para todo tiempo y recurso, se cumple φ_3. También sabemos que se cumple φ_4 dada la suposición inicial, es decir, si dos procesos son tales que $f(p) \cap f(p') \neq \emptyset$, entonces no estarán asignados al mismo tiempo. Entonces tenemos que φ es satisfacible por esa asignación (2)

Finalmente, por (1) y (2) se tiene que φ es satisfacible $\leftrightarrow$ existe una forma de planificar estos procesos. Por lo tanto, se puede resolver el problema planteado con un SAT solver.

Problema 2. Se dice que un conjunto de fórmulas Σ es finitamente satisfacible si cada subconjunto finito de Σ es satisfacible.

Demuestre que un conjunto de fórmulas es satisfacible si y sólo si es finitamente satisfacible.

Solución: Utilzaremos el siguiente lema:

Sea $\Sigma \subseteq L(P)$ finitamente satisfacible y $p \in P$. Entonces $\Sigma \cup \{p\}$ es finitamente satisfacible o $\Sigma \cup \{\neg p\}$ es finitamente satisfacible.

Ahora vamos a demostrar la dirección $\Leftarrow$ del teorema. La otra dirección es trivial. ($\Leftarrow$) Suponga que $P = \{p_i \mid i \geq 1\}$ y defina una sucesión $\{\Delta_i\}_{i\in\mathbb{N}}$ de conjuntos de literales: Caso base: $\Delta_0 = \emptyset$. Para $i \in \mathbb{N}: \Delta_{i+1} = \begin{cases} \Delta_i \cup \{p_{i+1}\} & \Sigma \cup \Delta_i \cup \{p_{i+1}\} \text{ es finitamente satisfacible} \\ \Delta_i \cup \{\neg p_{i+1}\} & \text{en caso contrario} \end{cases}$ Finalmente: $\Delta = \bigcup_{i\in\mathbb{N}} \Delta_i$

Para cada $p_i \in P : p_i \in \Delta_i \circ \neg p_i \in \Delta_i$, pero no ambas. Por lo tanto: Existe una única valuación σ que satisface a Δ. Vamos a demostrar que esta valuación satisface a Σ. Por contradicción: Suponga que $\sigma(\Sigma) = 0$. Entonces existe $\varphi \in \Sigma$ tal que $\sigma(\varphi) = 0$

Suponga que φ contiene n variables proposicionales y que p_k es la de mayor índice.

Tenemos que considerar dos casos. $\sigma(p_k) = 1$: entonces $\{\varphi\} \cup \Delta_{k-1} \cup \{p_k\}$ es inconsistente. Entonces: $\Sigma \cup \Delta_{k-1} \cup \{p_k\}$ no es finitamente satisfacible *y* $\neg p_k \in \Delta_k$. Contradicción: $\Delta_k \subseteq \Delta$ y $\sigma(p_k) = 1$. $\sigma(p_k) = 0$: entonces $\{\varphi\} \cup \Delta_{k-1} \cup \{\neg p_k\}$ es inconsistente. Entonces: $\Sigma \cup \Delta_{k-1} \cup \{p_k\}$ es finitamente satisfacible (por lema) y $p_k \in \Delta_k$ Contradicción: $\Delta_k \subseteq \Delta$ y $\sigma(p_k) = 0$.

Problema 3. Demuestre que $\{p, q \to (p \to r)\} \models q \to r$.

Solución: Sabemos que $\{p, q \to (p \to r)\} \models q \to r$ si y sólo si $\{p, q \to (p \to r), \neg(q \to r)\}$ es inconsistente. Para poder usar resolución, lo transformamos en un conjunto de cláusulas. Sean

$$\varphi = q \to (p \to r)$$
$$\psi = \neg(q \to r)$$

Usando las leyes de implicancia (dos veces) y asociatividad:

$$\varphi = q \to (p \to r) \equiv \neg q \vee (p \to r) \equiv \neg q \vee (\neg p \vee r) \equiv \neg q \vee \neg p \vee r$$

Usando las leyes de implicancia y De Morgan y la equivalencia entre conjunción y conjuntos:

$$\psi = \neg(q \to r) \equiv \neg(\neg q \vee r) \equiv q \wedge \neg r \equiv \{q, \neg r\}$$

Tenemos entonces el conjunto

$$\Sigma = \{p, \neg q \vee \neg p \vee r, q, \neg r\}$$

Basta demostrar que Σ es inconsistente, y como es un conjunto de cláusulas, lo haremos mostrando que $\Sigma \vdash \square$:

1. $p \quad \in \Sigma$
2. $\neg q \vee \neg p \vee r \quad \in \Sigma$
3. $\neg q \vee r$ resolución de $(1), (2)$
4. $q \quad \in \Sigma$
5. r resolución de $(3), (4)$
6. $\neg r \quad \in \Sigma$
7. $\square$ resolución de (5), (6)

Problema 4. Tenemos un grafo digirido G y la siguiente propiedad:

P: Si el grafo no tiene ciclos, entonces tiene un nodo que no tiene aristas entrantes (comúnmente llamada una raíz).

Hay que mostrar que P es verdad.

Decides hacerlo modelando cada grafo $G = (V, E)$, con $V = 1, \dots, n$, con un conjunto de proposiciones $P_E = \{e_{ij} \mid i, j \in \{1, \dots, n\}\}$, y las siguientes fórmulas: una fórmula $\varphi_E = \bigwedge_{(i,j)\in E} e_{i,j}$ y otra $\varphi_{\bar{E}} = \bigwedge_{(i,j)\notin E} \neg e_{i,j}$

1. Demuestra que existe una sola valuación que hace verdad a $\Sigma = \varphi_E, \varphi_{\bar{E}}$: la valuación que asigna un 1 a la variable e_{ij} si y solo si (i, j) es una arista en E. Esto nos va a permitir asumir que cada valuación para P_E corresponde a un grafo.

2. Construye una fórmula que sea verdad si y solo si el grafo G representado con las proposiciones y fórmulas tiene un nodo sin aristas entrantes.

Solución 1.

Vamos a asumir que existe una valuación τ que hace verdad Σ, es decir $\tau \models \Sigma$ y que está asigna un 1 aun $e_{i,j}$ que no es arista en E.

$$\begin{aligned} e_{i_0,j_0} &= 1 \\ e_{i_0,j_0} &\notin E \end{aligned}$$

Ahora podemos calcular las fórmulas usando lo definido anteriormente.

$$\begin{aligned} \varphi_E &= \bigwedge_{(i,j)\notin E} \neg e_{i,j} \\ &= \dots \wedge \neg e_{i_0,j_0} \wedge \dots = \dots \wedge 0 \wedge \dots = 0 \\ \varphi_{\bar{E}} &= 0 \to \tau \not\models \Sigma \end{aligned}$$

Llegamos a una contradicción porque tenemos que la valuación no satisface a Σ cuando habíamos supuesto que lo hacía. Por lo tanto queda demostrado que una valuación solo satisface a la fórmula si se asigna un 1 a un $e_{i,j}$ si y solo si $(i, j) \in E$.

Solución 2.

Una fórmula que es solo verdad si G tiene un nodo sin aristas entrantes es la siguiente:

$$\varphi_R = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{n} \neg e_{j,i} \right)$$

La parte de adentro de la paréntesis se asegura que no haya ningún nodo j tal que exista una arista que vaya de este al nodo i siendo examinado. Es una conjunción porque con solo una arista entrante ese nodo ya no es raíz. La. negación está ahí porque queremos que esta parte sea falsa si existe una arista. entrante. Fuera del paréntesis tenemos una disyunción porque basta con un solo nodo raíz para que sea verdad que G tiene un nodo sin aristas entrantes. Esta. además va a recorrer los nodos.