

# Ingeniería de Software II

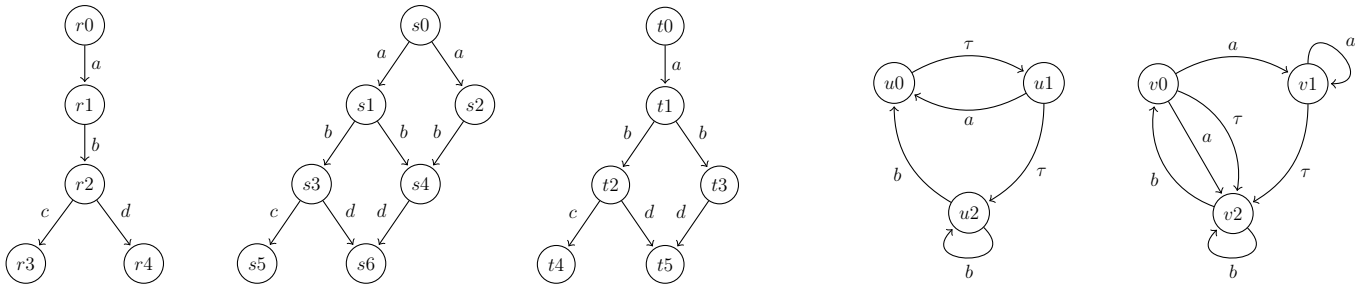


Figura 1

Ejercicio 1. Una relación  $R \subseteq S \times S$  es una *ready-simulación* si para todo  $(s, t) \in R$ ,

- $\forall s' \in S : s \xrightarrow{a} s' \Rightarrow \exists t' \in S : t \xrightarrow{a} t' \text{ y } (s', t') \in R$ , y además
- $(\exists t' \in S : t \xrightarrow{a} t') \Rightarrow (\exists s' \in S : s \xrightarrow{a} s')$ .

Considere los sistemas de transiciones etiquetadas de la Fig. 1.

- (a) Muestre que  $r_0$  y  $s_0$  son similares (es decir, que hay una simulación de  $r_0$  a  $s_0$  y otra de  $s_0$  a  $r_0$ ) pero que no son ready-similares.
- (b) Muestre que  $s_0$  y  $t_0$  son ready-similares (es decir, que hay una ready-simulación de  $s_0$  a  $t_0$  y otra de  $t_0$  a  $s_0$ ) pero que no son bisimilares.
- (c) Determine si existe una bisimulación débil entre  $u_0$  y  $v_0$ .

En todos los casos justifique su respuesta.

Ejercicio 2. Determine si los lenguajes definidos a continuación son de safety, liveness, ambos, o ninguno. Justifique su respuesta. En todos los casos considere el alfabeto  $\Sigma = \{a, b\}$ .

- (a) El lenguaje definido por el autómata de Büchi de la Fig. 2.
- (b) El lenguaje definido por la expresión  $\omega$ -regular  $(a^*bb)^*(a^\omega + b^\omega)$ .
- (c)  $P = \{\sigma \in \Sigma^\omega \mid \forall i \geq 0 : \#_a^i(\sigma) \geq \#_b^i(\sigma)\}$  donde  $\#_x^i(\sigma)$  es la cantidad de ocurrencias de la letra  $x$  en  $\sigma$  hasta la posición  $i$ , esto es

$$\#_x^i(\sigma) = \sum_{j=0}^i (\sigma(j) = x) ? 1 : 0.$$

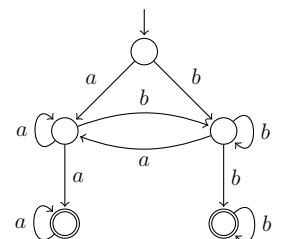


Figura 2

Ejercicio 3.

- (a) Dada dos fórmulas LTL  $\phi$  y  $\psi$ , el término  $\phi \frown \psi$  se interpreta como “Siempre ocurre  $\phi$  antes que  $\psi$ ”. Formalmente:

$$\sigma \models \phi \frown \psi \text{ sii } \forall i : i \geq 0 : \sigma[i..] \models \psi \Rightarrow \exists j : 0 \leq j < i : \sigma[j..] \models \phi$$

Dé una definición de  $\phi \frown \psi$  usando los operadores LTL y demuestre que su definición es correcta.

- (b) Considere un canal de una red LAN con broadcasting (tipo Ethernet u 802.11) con dos hosts conectados. Cada hosts  $H_i$  puede enviar mensajes al canal ( $i \in \{1, 2\}$ ) y esos mensajes pueden colisionar. Para representar estos eventos, considere, las proposiciones  $env\_msj_i$  y  $col\_det_i$ ,  $i \in \{1, 2\}$ , que denotan, respectivamente, el envío del mensaje y la detección de la colisión por parte del host  $H_i$ . Dé las fórmulas LTL que especifican que
- (I) siempre que los dos hosts envíen mensajes al canal simultaneamente, finalmente el host  $H_1$  detecta esa colisión;
  - (II) siempre que los dos hosts envíen mensajes al canal simultaneamente, el host  $H_1$  no envía un nuevo mensaje sin antes haber detectado la colisión.

**Ejercicio 4.** Determine si la siguiente afirmación es verdadera o falsa, y dé una explicación que justifique su respuesta:

“Un modelo  $M$  satisface la fórmula LTL  $\Box \Diamond p$  si y solo si se verifica que desde el estado inicial del grafo subyacente a  $M$  **no** se alcanza ningún ciclo tal que  $p$  no es válida en ninguno los nodos de dicho ciclo.”

**Ejercicio 5.** Considere una planta de producción de skates. Esta planta tiene dos robots de sujeción, dos robots de ensamble, una cinta transportadora intermedia y una cinta de salida, ambas de capacidad para  $N$  objetos. Los productos de entradas son las ruedas, los ejes y las tablas. Los robots de sujeción toman un objeto (ya sea un eje o una tabla, dependiendo de su función) y lo sujetan mientras esperan a que uno de los robots de ensamble ensamble las partes necesarias (ya sea las ruedas o los ejes ensamblados, según corresponda); una vez realizado los ensambles necesarios deja el objeto (los ejes ensamblados o el skate completo, respectivamente) en la cinta transportadora correspondiente. Los robots de ensamble toman los objetos a ensamblar (ruedas o ejes ensamblados) y los ensamblan en la pieza sostenida por el robot de sujeción correspondiente. De esta manera, cada robot de sujeción trabaja en conjunto con un robot de ensamble. El primer par de robots se encarga de ensamblar dos ruedas en los ejes (el robot de sujeción debe sostener el eje) y dejar el eje ensamblado en la cinta transportadora intermedia. El segundo par de robots se encarga de ensamblar dos ejes (con sus ruedas ensambladas) en la tabla (sujetada por el robot de sujeción) y dejarla en la cinta transportadora de salida.

- (a) Realice el el diagrama de estructura describiendo la arquitectura del modelo.
- (b) Modele el sistema usando FSP. Tenga en cuenta la simetrías que se presenta en algunas componentes. En particular, una cinta transportadora de capacidad  $N$  se puede modelar como:

```
CINTA(N=5) = COUNT[0],
COUNT[i:0..N] = (when (i<N) put->COUNT[i+1]
                  |when (i>0) get->COUNT[i-1]
                  ).
```

- (c) Dé las propiedades de progreso necesarias para asegurar que los robots de ensamble siempre están trabajando.