Modélisation et spécification – Master 2 LC

TD 5 : Automates étendus et protocoles

Mihaela Sighireanu

(`www.liafa.jussieu.fr/~sighirea/cours/modspec/`)

Protocole BRP

Le protocole BRP (*Bounded Retransmission Protocol*) est un protocole utilisé dans un des produits de Philips et assure un service de transmission de fichiers (FTS, *File Transmission Service*).

Spécification du service : Le FTS reçoit un fichier large de la part du client émetteur par le port S_{in}. Le fichier est représenté par une liste de petites parties $l = \langle d_1, ..., d_n \rangle$ avec $n > 0$. Le FTS délivre les données au client récepteur comme une liste de parties $e_1, ..., e_k$. Les parties peuvent être perdues, mais elles sont ni corrompues (erronées), ni reçues dans un autre ordre. Si une partie et perdue, la transmission est interrompue. Donc le client reçoit un prefix de la liste l. Les clients émetteur et récepteur reçoivent des indications sur le succès de l'opération de transfert. L'émetteur reçoit une indication par le port S_{out} et le récepteur reçoit une indication avec chaque partie. Le service est décrit logiquement par les propriétés ci-dessous.

S_{in}	:	$l = \langle d_1, ..., d_n \rangle \quad n > 0$
S_{out}	:	$i_s \in \{\texttt{I\_OK}, \texttt{I\_NOK}, \texttt{I\_DK}\}$
R_{out}	:	$\langle (e_1, i_1), ..., (e_k, i_k) \rangle \quad 0 \leq k \leq n$
		$i_j \in \{\texttt{I\_FST}, \texttt{I\_INC}, \texttt{I\_OK}, \texttt{I\_NOK}\} \quad 0 < j \leq k$
Pour $k > 0$		
(1.1)	:	$\forall 0 < j \leq k \ : \ i_j \neq \texttt{I\_NOK} \Longrightarrow e_j = d_j$
(1.2)	:	$n > 1 \Longrightarrow i_1 = \texttt{I\_FST}$
(1.3)	:	$\forall 1 < j < k \ : \ i_j = \texttt{I\_INC}$
(1.4.1)	:	$i_k = \texttt{I\_OK} \vee i_k = \texttt{I\_NOK}$
(1.4.2)	:	$i_k = \texttt{I\_OK} \Longrightarrow k = n$
(1.4.3)	:	$i_k = \texttt{I\_NOK} \Longrightarrow k > 1$
(1.5)	:	$i_s = \texttt{I\_OK} \Longrightarrow i_k = \texttt{I\_OK}$
(1.6)	:	$i_s = \texttt{I\_NOK} \Longrightarrow i_k = \texttt{I\_NOK}$
(1.7)	:	$i_s = \texttt{I\_DK} \Longrightarrow k = n$
Pour $k = 0$		
(2.1)	:	$i_s = \texttt{I\_DK} \Leftrightarrow n = 1$
(2.2)	:	$i_s = \texttt{I\_NOK} \Leftrightarrow n > 1$

Spécification (informelle) du protocole : Le protocole consiste d'un émetteur S ayant une horloge T_1 et un récepteur R ayant une horloge T_2 qui échangent des données via deux canaux (non fiables) K et L.

L'émetteur reçoit un fichier à transmettre et affecte le compteur de reprise (`rc`) à 0. Ensuite, il envoie à R les parties du fichier un par un sur K. L'horloge T_1 est mise à zéro et une trame est transmise par le canal K. Cette trame a la forme $(is\_first, is\_last, ab, data)$. Après avoir envoyé une trame, l'émetteur attend pour un acquittement ou pour une expiration de l'horloge T_1 (*timeout*). Si l'acquittement arrive, l'horloge T_1 est remise à zéro et soit le client émetteur est informé d'une transmission correcte, soit la partie suivante du fichier est transmise. Si l'horloge T_1 expire, la trame est retransmise (en incrémentant le compteur `rc`) ou la transmission est interrompue. La transmission est interrompue si la valeur du compteur `rc` dépasse la valeur `MAX`.

Le récepteur R attend la première trame. Cette trame est transmise au client récepteur, l'horloge T_2 est mise à zéro et l'acquittement est envoyé à S via L. Après, le récepteur attend l'arrivée des autres trames. Le récepteur mémorise si la trame est la dernière et la valeur du bit alterné. L'horloge T_2 est remise à zéro chaque fois qu'une nouvelle trame arrive. Elle expire si la transmission du fichier a été interrompue par l'émetteur.

Exercice 1 : Donnez la modélisation de la partie non temporisée du protocole BRP en utilisant les automates étendus (deux automates, un pour l'émetteur et un pour le récepteur).

Exercice 2 : Ajouter des horloges aux automates donnés afin de prendre en compte le comportement temporel du protocole.

Exercice 3 : Quelle est la relation entre les bornes de T_1 et T_2 pour que les expirations des horloges ne soient pas prématurées ?