

TP NE424 – Le protocole à fenêtre glissante

P Les textes présentés comme celui-ci repèrent des questions qui doivent être traitées avant la séance, pendant la **préparation** du TP.

Vous allez expérimenter avec le protocole à fenêtre glissante à l’aide d’un simulateur montrant l’interaction entre deux entités de couche liaison A et B.

Récupérez le simulateur (`swp.tgz`) depuis `chamilo` et mettez le en place à l’aide des commandes suivantes :

```
tar xvzf swp.tgz
cd SWP
make
```

Le protocole est réalisé dans le fichier `swp.c`, les autres fichiers source C sont ceux du simulateur proprement dit, ils ne nous intéressent pas.

La simulation se lance avec :

```
./swp durée tauxdeperte tauxderreurs trace duplex
```

où les paramètres sont les suivants :

<code>durée</code>	durée de la simulation en “top” d’horloges
<code>tauxdeperte</code>	pourcentage de trames perdues, entre 0 et 99
<code>tauxderreurs</code>	pourcentage de trames reçues avec une erreur détectée
<code>trace</code>	{ drapeaux pour tracer les événements (1 = émissions, 2 = réceptions, 4 = expirations, 8 = résumés périodiques)
<code>duplex</code>	0 pour une communication simplex

On conseille d’utiliser la valeur 7 (1+2+4) pour `trace`.

1 Prise en main

Cette partie n’a pas à être mentionnée dans le rapport.

1. Familiarisez-vous avec la sortie du simulateur. L’affichage est divisé principalement en 3 colonnes, de gauche à droite :
 - date, en tops d’horloge
 - événements concernant A :
 - `-->` émission correcte
 - `-->*` émission de trame qui est ensuite perdue
 - `<--` réception correcte
 - `*<--` réception avec *checksum* incorrect
 - événements concernant B (idem mais dans l’autre sens)

Pour chaque trame de données les trois champs indiqués sont dans l’ordre le numéro de séquence, le numéro d’acquittement *piggybacké* et les données transportées, indiquées entre guillemets. Dans le simulateur, les données sont simplement des nombres croissants, ce qui permet à la couche supérieure de vérifier que les données sont délivrées correctement. Si ce n’est pas le cas (protocole défectueux) la simulation s’interrompt en indiquant un échec du protocole.

Les expirations de temporisation sont également indiquées sur les côtés gauche et droit de l’écran.

2 Fenêtres et tampons

1. Faites un schéma montrant un extrait d'échange détaillé, en ajoutant l'état des fenêtres et le contenu des tampons (numéro de donnée, le cas échéant), en mode duplex.
2. Modifiez le code source de swp pour fixer la taille de la fenêtre d'émission (SWS) à 5 et de réception (RWS) à 4. Faites tourner le simulateur (en mode simplex) jusqu'à ce qu'un échec du protocole soit détecté. Expliquez en détail quel problème s'est produit. Faites là aussi un schéma détaillé des échanges complété par l'état des fenêtres et des tampons.
3. Modifiez à nouveau le code pour avoir SWS=3 et RWS=2. Expliquez précisément à l'aide d'un schéma pourquoi le problème ne peut plus se poser.

3 Performances

1. P En quoi est-il avantageux d'avoir RWS=SWS (plutôt que RWS=1 par exemple)? Quel peut être l'effet sur les performances d'augmenter encore RWS pour avoir RWS>SWS? Argumentez.
2. Avec SWS=4, évaluez l'impact de la taille de la fenêtre de réception RWS sur les performances du protocole. Vous pouvez par exemple tracer l'*efficacité* (définie comme le nombre de données délivrées sur le nombre de trames DATA transmises) en fonction du taux de perte et/ou d'erreur pour diverses valeurs de RWS. Commentez et surtout expliquez/interprétez les résultats.
L'efficacité est indiquée dans le rapport final affiché par le simulateur, elle est aussi fournie comme valeur de retour du processus (accessible par la variable `$?` dans le shell).
Le script shell `sim.sh`¹ montre comment vous pouvez réaliser de longues séries de mesures de manière automatique.
Pour tracer les courbes, utilisez le logiciel gnuplot.²
3. À part l'efficacité, quel critère de performance les tailles de fenêtres peuvent-elles influencer? Faites quelques mesures pour vérifier.

4 Un bug...

- P Le code de swp.c est légèrement différent de celui montré en cours. Expliquez, décrivez dans quelle situation le comportement diffère.

En fait celui montré en cours est erroné, au sens où la situation peut se trouver bloquée. Montrez un extrait d'échange où le problème est visible.

5 Modification

Dans cette partie, on s'intéresse à la mise en œuvre des acquittements négatifs (NAK).

1. En quoi l'ajout des NAK peut-il améliorer les performances?
2. Décrivez les modifications du protocole : les conditions qui déclenchent l'envoi d'un NAK, les actions à effectuer à la réception d'un NAK.
3. Décrivez comment vous implémentez ces modifications, puis comparez les performances avec la version initiale. Commentez.

1. Qui a quand même peut-être un petit bug facile à corriger...

2. Sous gnuplot, tapez `plot "monfichier" with linespoints` par exemple. Le fichier `graph.plt` fournit un exemple d'utilisation plus sophistiqué.