

Modélisation d'une connexion Compound TCP isolée

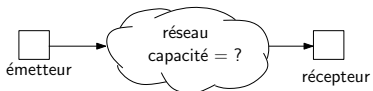
Alberto Blanc¹ Denis Collange¹ Konstantin Avrachenkov²

¹Orange Labs

²I.N.R.I.A. Sophia Antipolis

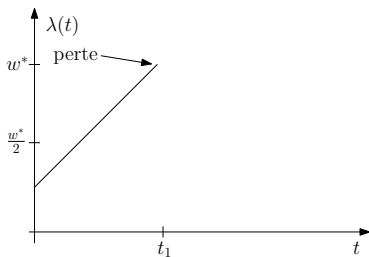
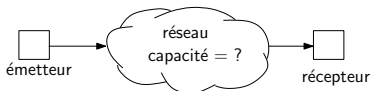
10 octobre 2008

TCP et le contrôle de congestion



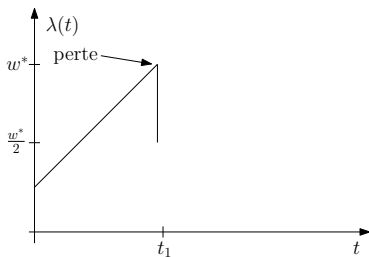
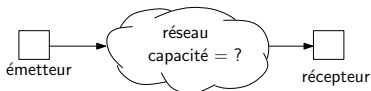
- quel débit ?

TCP et le contrôle de congestion



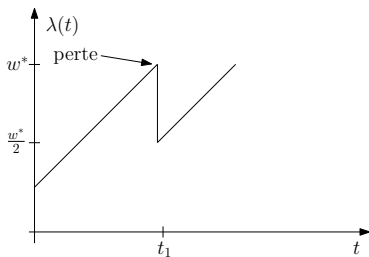
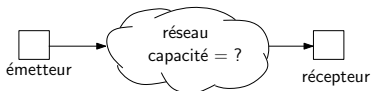
- quel débit ?
- augmenter le débit jusqu'à une perte

TCP et le contrôle de congestion



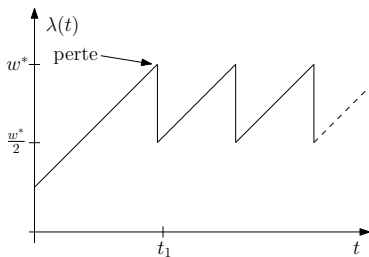
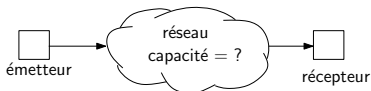
- quel débit ?
- augmenter le débit jusqu'à une perte
- diminuer le débit de moitié

TCP et le contrôle de congestion



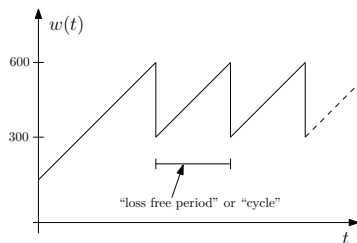
- quel débit ?
- augmenter le débit jusqu'à une perte
- diminuer le débit de moitié
- recommencer la phase de croissance

TCP et le contrôle de congestion



- quel débit ?
- augmenter le débit jusqu'à une perte
- diminuer le débit de moitié
- recommencer la phase de croissance
- c'est la "congestion avoidance"

Limitations de TCP

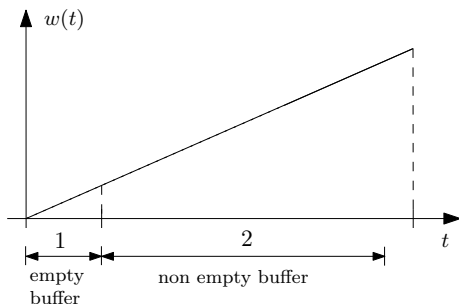


- $\lambda = \frac{w}{\tau}$ (τ =Round Trip Time)
- produit délai-bande-passant : "quantité de paquets qu'on peut stocker dans le tuyau"
- large produit délai-bande-passant \Rightarrow longs cycles
- TCP "traditionnel" (NewReno) pas idéal pour réseaux haut débit

Motivations

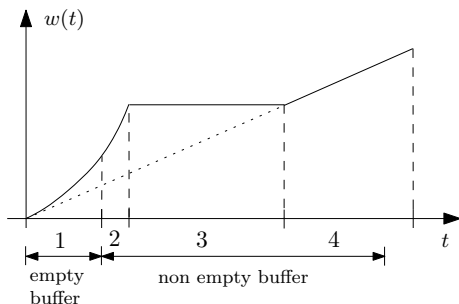
- mise en œuvre sur Microsoft Vista
- pas de modèle analytique (complet)
- commencer avec le cas le plus simple (connexion isolée)
- même ce cas n'est pas si simple
- validation de la mise en œuvre

La fenêtre de Compound TCP



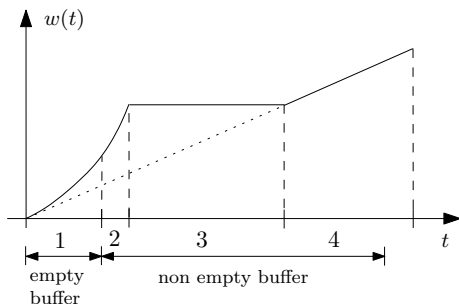
- Reno : $w_{i+1} = w_i + 1$ (chaque RTT)

La fenêtre de Compound TCP



- Reno : $w_{i+1} = w_i + 1$ (chaque RTT)
- Compound (chaque RTT) :
 - $w_{i+1} = w_i + \alpha w_i^k$ (phases 1,2)
 - $w_{i+1} = w_i$ (phase 3)
 - $w_{i+1} = w_i + 1$ (phase 4)

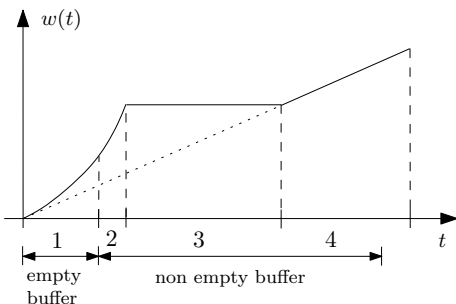
La fenêtre de Compound TCP



- Reno : $w_{i+1} = w_i + 1$ (chaque RTT)
- Compound (chaque RTT) :
 - $w_{i+1} = w_i + \alpha w_i^k$ (phases 1,2)
 - $w_{i+1} = w_i$ (phase 3)
 - $w_{i+1} = w_i + 1$ (phase 4)

- Δ_i : estimation de la taille de la file d'attente (une fois par RTT)
- si $\Delta_i > \gamma$ passer de la phase 2 à 3

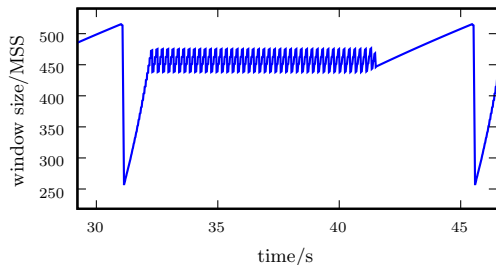
La fenêtre de Compound TCP



- Reno : $w_{i+1} = w_i + 1$ (chaque RTT)
- Compound (chaque RTT) :
 - $w_{i+1} = w_i + \alpha w_i^k$ (phases 1,2)
 - $w_{i+1} = w_i$ (phase 3)
 - $w_{i+1} = w_i + 1$ (phase 4)

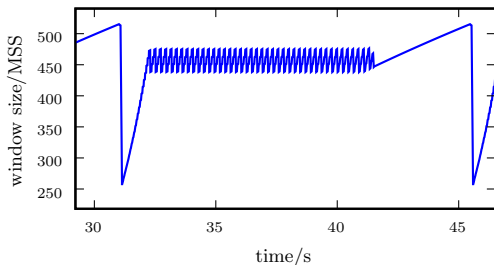
- Δ_i : estimation de la taille de la file d'attente (une fois par RTT)
- si $\Delta_i > \gamma$ passer de la phase 2 à 3
- durée de la phase "fenêtre constante" dictée par la dynamique de Reno

Cela était la théorie ...



- w n'est pas constant (une connexion, pas d'autre trafic, $\mu = 100$ Mb/s, $RTT = 50$ ms)

Cela était la théorie ...

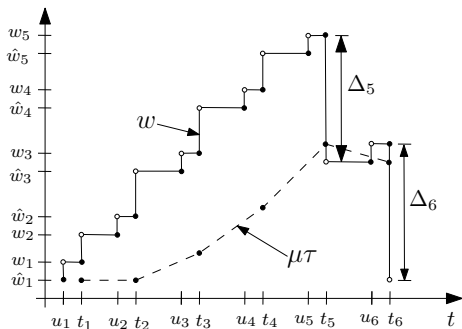
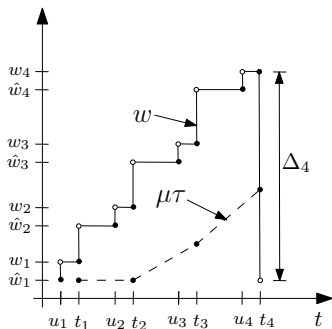


- w n'est pas constant (une connexion, pas d'autre trafic, $\mu = 100$ Mb/s, $RTT = 50$ ms)
- pendant la phase de la "fenêtre constante ."

$$w_{i+1} = \begin{cases} w_i + \alpha w_i^k & , \text{si } \Delta_i < \gamma \\ w_i - \zeta \Delta_i + 1 & , \text{si } \Delta_i \geq \gamma \end{cases}$$

- $\alpha = \frac{1}{8}$, $k = \frac{3}{4}$, $\gamma = 30$, $\zeta = 1$ (comme dans les papiers décrivant Compound TCP)

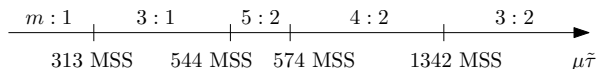
Différents types d'oscillations (phase 3) (1)



- $m:n$ cycles (m augmentations suivie par n réductions)
- on a observé les cycles 6:1, 5:1, 4:1, 3:1 et 5:2, 4:2, 3:2 (simulations)

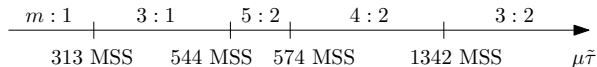
Différents types d'oscillations(phase 3) (2)

- type d'oscillations dépend du produit délai-bande-passante $\gamma = 30$:

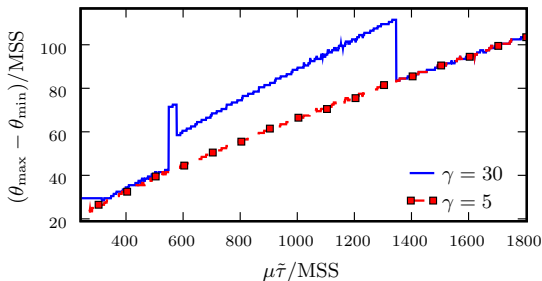


Différents types d'oscillations(phase 3) (2)

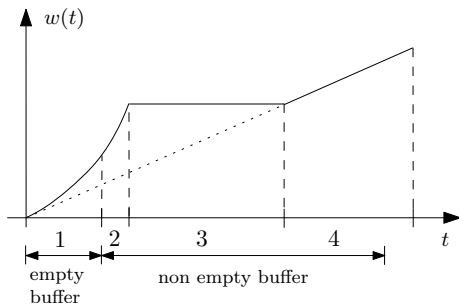
- type d'oscillations dépend du produit délai-bande-passante $\gamma = 30$:



- possible de calculer la taille et le point de départ des oscillations (équations de point fixe)
- taille augmente avec le produit délai-bande-passante

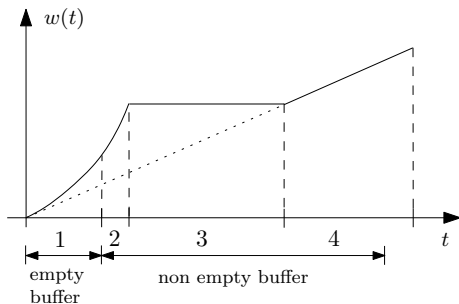


Évolution de la fenêtre de transmission (Sending Window)



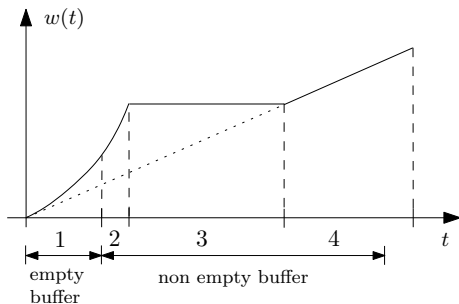
- comment peut-on combiner les 4 phases de la fenêtre ?

Évolution de la fenêtre de transmission (Sending Window)



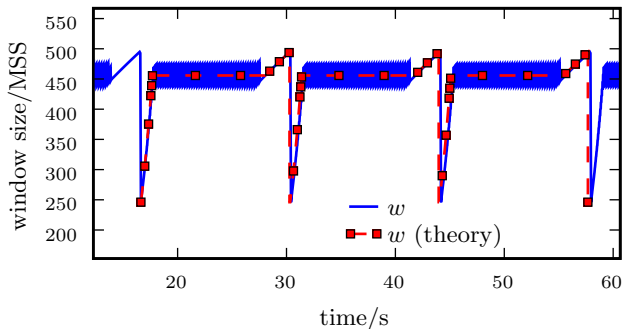
- comment peut-on combiner les 4 phases de la fenêtre ?
- 6 cas possibles :
 - 1 seulement phase 1
 - 2 phases 1,2
 - 3 seulement phase 2
 - 4 phases 1,2,3, 4
 - 5 phases 2,3,4
 - 6 seulement phase 4

Évolution de la fenêtre de transmission (Sending Window)



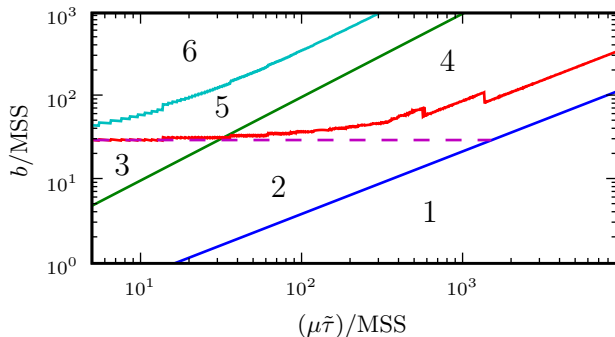
- comment peut-on combiner les 4 phases de la fenêtre ?
- 6 cas possibles :
 - 1 seulement phase 1
 - 2 phases 1,2
 - 3 seulement phase 2
 - 4 phases 1,2,3, 4
 - 5 phases 2,3,4
 - 6 seulement phase 4
- deux "cas extrêmes" (seulement phase 1 ; seulement phase 2)

Exemple : cas 4



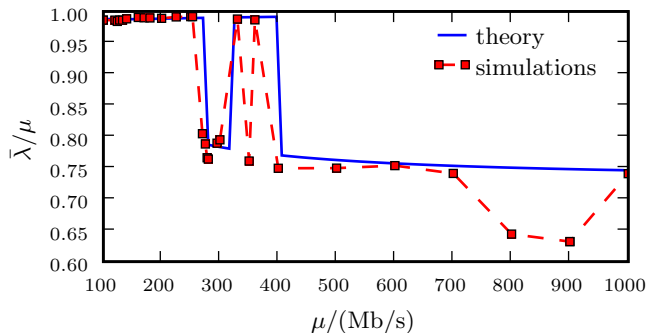
- modèles analytiques différents pour chaque phase
- peuvent être combinées pour les différents cas :
 - bonne correspondance avec les simulations pour les cas 4,5 et 6
 - pas aussi bonne pour les cas 1,2 et 3

Sélection du cas pour l'évolution de la fenêtre



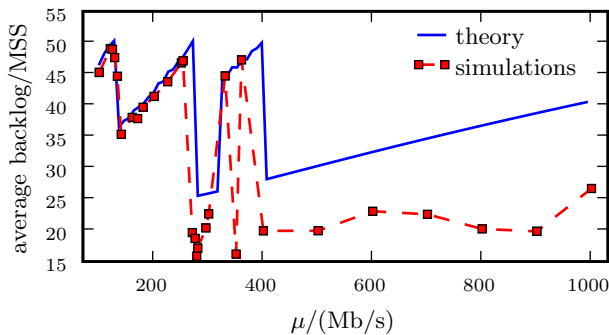
- dépend du produit délai-bande-passante *et* de la taille du buffer
- oscillations pendant la phase 3 ont un impact non négligeable (ligne pointillée)
- approximations en nombres entiers du noyau Linux explicitement modélisés

Débit normalisé



- bonne correspondance pour les petites valeurs du produit délai-bande-passante
- transitions à 275, 320, 400 Mb/s causées par les oscillations pendant la phase 3

Taille moyenne de la file d'attente



- bonne correspondance pour les petites valeurs du produit délai-bande-passante
- taille moyenne de la file d'attente change même lorsque le lien est pleinement utilisées
- transitions causées par les oscillations pendant la phase 3 comme pour le débit

Conclusions

- modèle détaillé d'une connexion Compoun TCP isolée
- comportement complexe (même pour une connexion isolée)
- oscillations non négligeables pendant la phase 3
 - débit plus faible
 - impact sur les autres connexions ?