



HAL
open science

L'outil Réseaux de Petri hybrides dans les réseaux de communication : dynamique des transmissions et étude de comportement

Melha Bitam, Hassane Alla

► **To cite this version:**

Melha Bitam, Hassane Alla. L'outil Réseaux de Petri hybrides dans les réseaux de communication : dynamique des transmissions et étude de comportement. *Journal Européen des Systèmes Automatisés (JESA)*, 2006, 40, pp.73-94. hal-00109310

HAL Id: hal-00109310

<https://hal.science/hal-00109310>

Submitted on 24 Oct 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

L'outil Réseaux de Petri hybrides dans les réseaux de communication : dynamique des transmissions et étude de comportement.

Melha Bitam* — Hassane Alla*

* Laboratoire d'Automatique de Grenoble

CNRS – INPG – UJF

941 rue de la Houille Blanche

B.P. 46 Saint Martin d'Hères cedex

Melha.bitam@lag.ensieg.inpg.fr

Hassane.alla@lag.ensieg.inpg.fr

RÉSUMÉ. Afin d'évaluer l'influence des réseaux de communication dans la transmission des données, nous nous proposons dans ce papier de modéliser d'abord puis d'étudier les dynamiques et les performances d'une ligne de transmission TCP/IP soumise aux protocoles de contrôle de congestion et comprise dans un environnement Internet. En effet, l'utilisation très forte des réseaux de communications et notamment les réseaux TCP/IP fait que ces derniers sont constamment très fortement chargés et les transmissions résultantes sont soumises à des problèmes de pertes de données (ou congestion). L'outil de modélisation utilisé est les réseaux de Petri hybrides. L'outil de simulation est Matlab.

ABSTRACT. In order to evaluate the influence of communication networks in data transmission, we propose in this work to model and then to study performances and behaviours of TCP/IP transmission line submitted to congestion avoidance protocols and merged in Internet environment. In fact, with the growth of communication networks, transmission lines were strongly charged and severe congestion problems appears generating data losses. Modeling tool used is hybrid Petri nets. Simulation tool is Matlab.

MOTS-CLÉS : Modélisation, Evaluation de performances, réseaux de Petri hybrides, réseaux TCP/IP et contrôle de congestion, Simulation.

KEYWORDS: Modelisation, Performances evaluation, Hybrid Petri nets, TCP/IP networks and congestion control, Simulation.

1. Introduction

Les réseaux de communication offrent à ce jour un ensemble de services très diversifié aussi bien pour les industriels, les banques, la recherche, que dans la

médecine et la commande à distance. A cette forte utilisation des réseaux viennent s'ajouter des contraintes de performances, de sécurité et de qualité de service que les fournisseurs de service se doivent de respecter. L'étude des performances des réseaux ainsi que leur modélisation exacte est donc un pas important pour la connaissance des influences des différents liens des lignes de transmission sur les données qui les traversent. Ce travail s'insère dans le cadre du projet Network Controlled System (NECS, projet CNRS) qui traite de problèmes où le contrôle interagit avec la théorie de l'information. Les travaux menés dans le cadre de ce projet visent à évaluer de nouveaux problèmes de contrôle posés entre autres par la considération de nouveaux composants technologiques et de l'augmentation de la complexité des systèmes.

Nous nous sommes penchés, pour notre part, sur les réseaux TCP/IP car se sont les plus utilisés actuellement dans Internet. Ces réseaux sont dits fiables, i.e., ils n'admettent pas de pertes de données. La mise en oeuvre de cette qualité de service se fait par l'acquiescement des données bien reçues.

Alors qu'il est clairement recommandé de proposer des modèles complets qui puissent faire ressortir les différentes dynamiques d'une ligne de communication, nous n'avons pas trouvé de tels modèles dans nos lectures. Nous avons donc étudié les dynamiques et les protocoles d'une ligne TCP/IP soumise à l'environnement extérieur Internet et nous avons établi un modèle à base duquel toutes les dynamiques peuvent être observées à tous les niveaux d'une communication (charge des buffers, baisse des vitesses à l'entrée des routeurs, séparation des flux par les routeurs, priorités, stockage des données, retards, pertes, etc.) (Bitam *et al.*, 2005). Ce modèle détaillé nous a aussi permis de visualiser directement l'effet des changements de paramètres de la ligne et des protocoles sur l'évolution de la transmission sur la ligne: Un large panel de simulations a donc été effectué afin de visualiser ces différentes performances. D'une part, le modèle établi permet de visualiser directement la quantité de pertes produites sur la ligne durant une transmission. D'autre part, les protocoles de communication, à partir des données envoyées et celles reçues, déduisent la quantité de donnée perdue. Nous disposons ainsi d'une comparaison pertes réelles - pertes vues par les protocoles qui peut servir d'évaluation des protocoles mêmes. Plus les deux valeurs des pertes sont proches, plus les protocoles sont performants.

Les systèmes de communications comportent des événements continus aussi bien que discrets. Des flux arrivent au récepteur selon des dynamiques continues. Ces dynamiques sont modifiées ponctuellement par des commandes discrètes provenant des protocoles de contrôle. Ce comportement définit parfaitement celui d'un système hybride. Parmi les outils de modélisation des systèmes hybrides, il y a ceux qui représentent une extension des modèles continus, comme les bonds graph, ceux qui sont une extension des modèles discrets comme les réseaux de Petri hybrides (RdPH), et enfin les modèles mixtes qui combinent les deux parties discrète et continue comme les RdP mixtes et des statecharts mixtes (Zaytoon, 2001). D'une part, nous nous intéressons à un domaine très particulier constitué par les réseaux de communication. Il n'est donc pas dans notre intérêt d'utiliser un formalisme trop général. D'autre part, nous désirons établir un modèle lisible et compréhensif qui

mette en évidence les différentes parties d'un réseau. Pour cela, notre choix s'est porté sur les réseaux de Petri hybrides comme outils de modélisation des réseaux de communication (Bitam, 2004}. Il s'est avéré que cet outil nous permet de modéliser les réseaux en tenant compte de toute la complexité des différentes dynamiques de tels systèmes (flux de message et protocoles de contrôle).

Ce papier est partagé en 6 sections. La section 2 concerne les protocoles de contrôle au sein des réseaux TCP/IP. Dans la section 3, on présentera l'outil de modélisation réseaux de Petri hybride. La modélisation continue et discrète d'une ligne de transmission soumis aux protocoles du TCP et baignée dans un environnement extérieur Internet fera l'objet de la 4ème section. Dans la section 5 sont présentés les résultats de simulation. Ceux-ci permettent en même temps de valider le modèle et d'évaluer les performances du système. Nous terminons ce travail par quelques conclusions et perspectives.

2. Algorithmes de contrôle de congestion

2.1. Algorithmes slow start et congestion avoidance :

L'algorithme slow start est utilisé lorsqu'on démarre une transmission ou suite à des pertes de données. On introduit deux variables appelées *cwnd* et *ack*. *Cwnd* (congestion window) va représenter la taille de fenêtre des données envoyées. Au départ, *cwnd* est initialisée à un segment. Si *cwnd* est bien reçue par le récepteur, celui-ci envoie un acquittement *ack* à l'émetteur signifiant que la transmission est réussie. La prochaine taille de *cwnd* sera alors de deux segments. Cette nouvelle fenêtre est envoyée sur la ligne. A la réception de chaque acquittement, la nouvelle fenêtre va augmenter de 2 segments. A la réception des 2 acquittements, la taille de la fenêtre *cwnd* sera donc de 4 segments, etc. Pendant la phase slow start, la fenêtre *cwnd* augmente de façon exponentielle (Figure 1.a) (Jacobson, 1988 ; Bitam, 2004). Lorsque la taille de la fenêtre *cwnd* atteint un seuil appelé seuil d'évitement de congestion (slow start threshold), *ssth*, l'algorithme appliqué pour la transmission des données est alors congestion avoidance. Pendant cette phase, l'évolution de la taille de *cwnd* est linéaire (Figure 1.b). En effet, si par exemple, la taille de *cwnd* est de 9 segments et si les 9 *ack* correspondants sont arrivés à l'émetteur, la nouvelle taille de fenêtre envoyée sera égale à 10 segments.

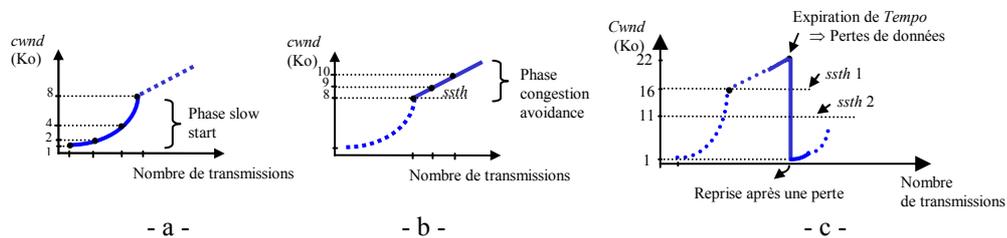


Figure 2. a) Phase exponentielle, b) Phase linéaire, c) Reprise après pertes.

2.2. Temporisateur de retransmission *Tempo* et son calibrage

Lorsqu'une donnée est envoyée, l'émetteur attend l'arrivée de l'acquittement pendant un temps bien défini. Celui-ci est donné par un temporisateur appelé temporisateur de retransmission (retransmit timer), *Tempo*. A l'envoi d'une donnée, ce temporisateur est enclenché. Si, à échéance de *Tempo*, l'acquittement de la donnée envoyée n'est pas arrivé, celle-ci est considérée comme perdue. Il y a congestion. Le calibrage de *Tempo* est de ce fait très important. Il a été élaboré par Jacobson ([13]) de façon à ce que *Tempo* ne soit ni trop courte ni trop longue. En effet, si *Tempo* est trop courte, on pourrait déduire une congestion prématurément et recevoir les acquittements des données un moment après. Si, par contre *Tempo* est trop longue, on risque de réagir tardivement aux pertes. Temps pendant lequel les données continuent de se perdre. Finalement, *Tempo* est calculée selon la valeur du temps de boucle sur une ligne (*RTT*, Jacobson, 1988 et Tanenbaum, 1997) puis mis à jour à chaque réception d'un acquittement.

2.3. Réaction aux pertes

Lorsque des données se perdent, i.e., lorsque *Tempo* expire, on diminue brutalement la taille de la fenêtre *cwnd*. Celle-ci est réinitialisée à la taille d'un segment (m octets) avant de reprendre la transmission suivant l'algorithme slow start.

Notons que les algorithmes slow start et congestion avoidance sont différents et sont implantés séparément. Toutefois, la transmission se fait selon l'algorithme slow start jusqu'à atteindre le seuil *sssth* (slow start threshold), puis se poursuit selon l'algorithme congestion avoidance. Lorsque ces pertes se produisent, nous revenons à nouveau à slow start (Figure 1.c).

3. Outils de modélisation: Les réseaux de Petri hybrides

Les Réseaux de Petri (RdP) sont un outil qui permet de modéliser de façon simple et intuitive des systèmes de nature très variée. Les RdP sont de nature essentiellement discrète. En effet, ils permettent de modéliser des états discrets d'un système tel que la marche ou l'arrêt d'une machine. Cependant, ces modèles discrets se sont rapidement vus limités car ils explosent rapidement lorsque le nombre d'états croît. Cette réalité a engendré l'apparition des RdP continus {Alla *et al.*, 1998}. Dans les RdP continus, des quantités réelles sont associées aux places et des vitesses de franchissements sont associées aux transitions. Les RdP hybrides contiennent au moins un nœud (place ou transition) discret et au moins un nœud continu.

3.1 Quelques règles et définitions des RdP continus temporisés

Soit un RdP R continu à vitesses constantes.

DEFINITION 1. — Une place P_i de R est **alimentée** s'il y a au moins une transition T_j parmi les transitions d'entrées de P_i qui est validée.

DEFINITION 2. — Une place T_j est **validée** si chaque place P_i d'entrée de T_j satisfait au moins une des deux conditions suivantes:

- $m_i(t) > 0$
- P_i est alimentée

REMARQUE 1. — Une transition est **fortement validée** si toutes ses places d'entrées satisfont la première condition, autrement elle est **faiblement validée**.

Vitesse de franchissement d'une transition Notons:

$$I_i = \text{Pré}(P_i) = \sum_{T_k \in {}^\circ P_i} \text{Post}(P_i, T_k) \cdot v_k = \text{vitesse d'alimentation de la place } P_i$$

$$O_i = \text{Post}(P_i) = \sum_{T_k \in P_i^\circ} \text{Pré}(P_i, T_k) \cdot v_k = \text{vitesse de vidage de la place } P_i.$$

L'écart $I_i - O_i$, noté B_i , représente le bilan dynamique de la place P_i . Le bilan B_i peut être positif, négatif ou nul.

PROPRIETE 1. — La vitesse de franchissement d'une transition T_j *fortement validée* est égale à sa vitesse maximale de franchissement V_j . La vitesse de franchissement d'une transition T_j *faiblement validée* est donnée par la relation suivante:

$$v_j(t) = \min[V_j, \min_i (B_i + v_j(t))], \quad i \text{ tel que } P_i \text{ une place d'entrée } T_j \text{ et } m_i(t) = 0.$$

Le marquage 0^+ : Dans les RdP discrets, un test à zéro est simple à réaliser car une place vide est une place qui contient moins d'un jeton. Un arc inhibiteur dont le poids est 1 permet ainsi d'effectuer ce test. Si l'on veut introduire la notion de test à zéro pour les RdP continus, le problème est autre. En effet, une transition continue est validée si sa place en amont possède un marquage réel strictement positif. Or, il n'y a pas de marquage réel positif au dessous duquel on peut dire qu'un marquage continu est nul. La notion du 0^+ a alors été introduite. Elle représente une quantité infiniment petite mais non nulle {David *et al.*, 2001}. L'association du poids 0^+ à un arc inhibiteur permet d'effectuer un test à zéro d'une place continue.

REMARQUE 2. — Le marquage d'une place P noté \tilde{m} signifie que ce marquage est un nombre non standard, ici 0^+ .

3.2. Fonctionnement d'un RdP à vitesses constantes: RdPCC

Nous allons d'abord présenter de manière intuitive le fonctionnement d'un RdPCC. L'exemple du système à réservoir de la figure 2.a comprend deux réservoirs et une pompe (Bitam *et al.*, 2003). Le réservoir 1 se vide dans le second par l'effet de la gravité à la vitesse V_1 . Le réservoir 2 se vide à la vitesse V_2 grâce à une pompe P qui à son tour remplit le réservoir 1 et ainsi de suite. Ce système est continu et peut être représenté par le RdP continu de la figure 2.b. Deux places continues représentent les réservoirs 1 et 2. Le marquage de ses places indique la quantité d'eau initiale

présente dans les réservoirs. Les vitesses V_1 et V_2 sont associées aux transitions T_1 et T_2 . Dans un premier temps, la place P_1 se remplit à la vitesse V_2 et se vide à V_1 . La place P_2 a exactement la dynamique opposée. D'où les équations :

$$m_1 = V_2 - V_1; \tag{1}$$

$$m_2 = V_1 - V_2. \tag{2}$$

et

$$m_1(t + dt) = m_1(t) + (V_2 - V_1)dt; \tag{3}$$

$$m_2(t + dt) = m_2(t) + (V_1 - V_2)dt. \tag{4}$$

m_1 et m_2 étant les marquages des places P_1 et P_2 respectivement.

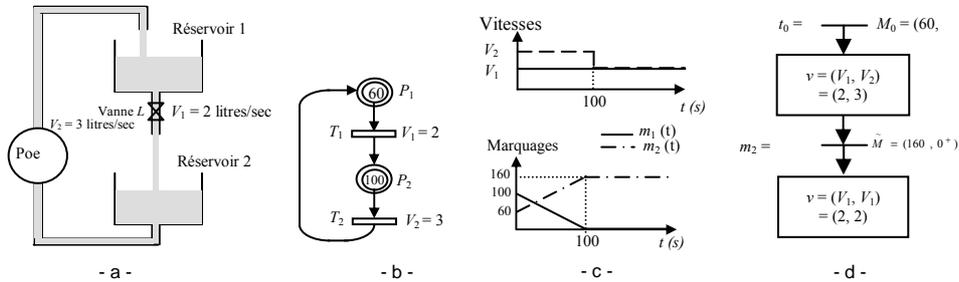


Figure 1. a) système à réservoir d'eau, b) RdP continu équivalent, c) Dynamique des marquages et évolution de vitesses. d) Graphe d'évolution d'un RdPCC

Les vitesses $v_1(t)$ et $v_2(t)$ représentent les vitesses instantanées de franchissement des transitions T_1 et T_2 . Elles ont pour valeurs maximales V_1 et V_2 . Les vitesses associées aux transitions d'un RdP continu sont les vitesses maximales (Figure 2.b).

D'une part, selon que V_1 est supérieure à V_2 ou inversement, un des marquages, m_1 ou m_2 va s'annuler (équations 3 et 4). D'autre part, Un marquage ne peut pas être négatif. Il s'ensuit qu'au moment du passage d'un marquage à zéro, les dynamiques vont changer.

Les évolutions données par la figure 2.c sont obtenus en prenant $V_1 = 2$ l/sec, $V_2 = 3$ l/sec, $m_1(0) = 60$ litres et $m_2(0) = 100$ litres. Le comportement possède donc 2 phases. Une première phase pour $t \in [0,100]$ où les vitesses sont constantes égales à $(v_1, v_2) = (V_1, V_2)$ et où l'évolution des marquages se fait selon les équations 3 et 4. Une deuxième phase où le vecteur de vitesses est constant à $(v_1, v_2) = (V_1, V_1)$, la dynamique des marquages devenant alors nulle. Les vitesses sont constantes par intervalle de temps, elles changent lors du passage d'un marquage à zéro. La dynamique des marquages de ce système est donc **constante par morceaux**.

3.3. Graphe d'évolution d'un RdPCC

Le nombre de marquages d'une place étant infini dans les RdP continus, on ne peut pas établir de graphe de marquages. Cependant, les vecteurs vitesses instantanés de

franchissement restent constants dans des intervalles donnés. De ce fait, nous pouvons nous baser sur les vecteurs de franchissement ainsi que sur les vecteurs de marquages du RdP à chaque changement de dynamique pour définir complètement l'évolution de ce RdP. On construit donc un graphe d'évolution comme un RdP où chaque place représente un **IB-état** et où à chaque transition est associé l'événement qui provoque le changement de l'état des vitesses (un changement de marquage). Un **IB-état** est une phase d'évolution du RdP où *le vecteur de vitesses instantanées reste constant*.

Déterminons le graphe d'évolution correspondant au RdP de la Figure 2.b. Le marquage initial est $M_0 = (60, 100)$ et à cet instant $v = (V_1, V_2) = (2, 3)$ (Figure 2.c). Ce vecteur vitesse restera constant jusqu'à l'instant $t_1=100s$ où m_2 atteint la valeur zéro (noté $m_2 = 0$ devant la transition correspondante). L'instant d'occurrence de cet événement est noté après la barre /. Ce marquage nul va entraîner un changement du vecteur vitesses à $v = (V_1, V_1) = (2, 2)$. Lorsque tous les bilans dynamiques B_i sont positifs ou nuls, le graphe d'évolution est obtenu.

3.4. Les RdP hybrides

Un RdP hybride contient au moins un nœud continu et au moins un nœud discret. La modélisation du temps dans les RdP hybrides est détaillée dans (David *et al.*, 2001 et David *et al.*, 2005) et consiste à considérer des vitesses de franchissement sur les transitions continues (*C-transitions*) et des délais de franchissement sur les transitions discrètes (*D-transitions*).

DEFINITION 3. — Une transition continue (**C-transition**) T_j est **validée** si chacune de ses places d'entrée P_i satisfait les conditions suivantes:

Si P_i est une place discrète : $m_i \geq \text{Pré}(P_i, T_j)$,

Si P_i est une place continue : $-m_i > 0$,

$-P_i$ est alimentée, i.e., il existe au moins une transition

d'entrée de P_i pour laquelle la vitesse de franchissement est non nulle.

Une *C-transition* T_j est dite *fortement validée* si $m_i > 0$ pour chacune de ses places d'entrées P_i , sinon elle est dite *faiblement validée*.

DEFINITION 4. — On dit d'une transition discrète (**D-transition**) T_j qu'elle est **validée** si chacune de ses places d'entrées P_i satisfait la condition: $m_i \geq \text{Pré}(P_i, T_j)$.

3.4.1. Graphe d'évolution d'un RdP hybride

Un graphe d'évolution d'un RdP hybride est constitué d'une succession d'IB-états représentés par des noeuds (Figure 3). On passe d'un noeud à un autre par l'occurrence d'un des événements suivants:

- Le marquage d'une *C-place* s'annule (appelé C1-événement),
- Le franchissement d'une *D-transition* (appelé D1-événement),

– Le degré de validation d'une D-transition qui change à cause du marquage d'une C-place (appelé D2-événement).

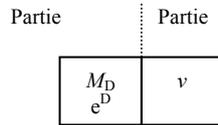


Figure 3. Représentation d'un IB-état

Chaque noeud comprend une partie discrète où sont indiqués le marquage total M_D et le vecteur de validation des D -places e^D une partie continue où est indiqué le vecteur vitesses instantanées v .

4. Le modèle d'une ligne de communication TCP/IP

Nous allons étudier l'exemple d'une ligne de transmission composée d'un émetteur E et d'un récepteur R reliés par un routeur Rt et deux canaux de transmission ayant les débits respectifs V_1 et V_2 . Le routeur possède un buffer de capacité maximale notée C . Au delà de cette valeur, le buffer ne peut plus recevoir les données et celles-ci sont perdues (Figure 4).

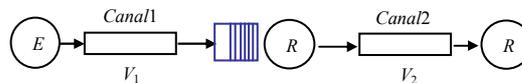


Figure 4. Composition globale de la ligne de communication considérée.

Deux parties distinctes vont apparaître dans la modélisation de cette ligne. L'une continue qui représente le flux de messages sur la ligne et l'autre, discrète, qui représente les consignes discrètes des protocoles. Nous sommes donc en présence de deux dynamiques, l'une discrète et l'autre continue qui interagissant entre elles pour donner une dynamique continue ponctuellement modifiée par une dynamique discrète. Ceci est la définition même d'un système hybride continu-discret.

4. 1. Le modèle continu d'une ligne de communication TCP/IP

Nous allons commencer par exposer le modèle d'une ligne de communication sans l'environnement extérieur Internet (Figure 5) (Bitam *et al.*, 2005). Cette ligne comprend la modélisation du routeur, du buffer et des canaux de transmission. Elle comprend aussi les pertes occasionnées lorsque le buffer est plein.

– Le modèle du routeur : Un routeur se comporte comme un réservoir à capacité limitée. Les données arrivent à la vitesse V_1 et sortent à la vitesse V_2 . Ainsi, la place P_4 la Figure 5 représente la charge du routeur et la place complémentaire P_5 représente l'espace libre restant dans le buffer. La somme $m_4 + m_5$ reste toujours constante égale à la capacité maximale du routeur C . C'est un invariant de marquage.

– Le modèle du canal de transmission : Un canal de transmission transmet des données mais ne les stocke pas. Il se comporte donc comme un réservoir à capacité nulle. Pour cela, on va utiliser le nouveau concept du 0^+ , introduit par David et Alla dans (David *et al.*, 2001). Un marquage 0^+ associé à une place continue indique une quantité infiniment petite mais non nulle. Un tel marquage permet de valider chacune des deux transitions T_1 et T'_1 et d'assurer que la vitesse d'entrée du canal est toujours égale à sa vitesse de sortie.

– Modélisation des pertes dans une ligne de communication : Lorsque le buffer se charge, il ne peut plus recevoir de données à la vitesse V_1 et sa vitesse d'entrée instantanée $v_1(t)$ s'aligne à V_2 . Ce débit d'entrée vers la place Rt représente ce que le routeur *peut* prendre comme données. Cependant, l'émetteur continue à émettre à la vitesse V_1 et la différence entre ce qui arrive au routeur ($v'_1(t)$) et ce qui est réellement pris par ce dernier ($v_1(t)$) est perdu. C'est le phénomène de congestion. Les pertes peuvent être modélisées comme indiqué sur la figure 5 où une transition, T''_1 , de priorité inférieure à T_1 , est ajoutée à la sortie du canal 1. Le marquage 0^+ d'une place assure qu'à tout instant la somme de ce qui entre dans cette place est égale à la somme de ce qui en sort. Ainsi, $v_1(t) = v_1(t) + v''_1(t)$ est toujours vrai. Le modèle de la ligne de communication soumise à des pertes est représenté par la figure 5. Ce modèle comprend, en plus des modèles du routeur et des canaux (places Rt , $Cnl1$ et $Cnl2$), les places E et R qui représentent la quantité de données présentes dans l'émetteur et le récepteur respectivement.

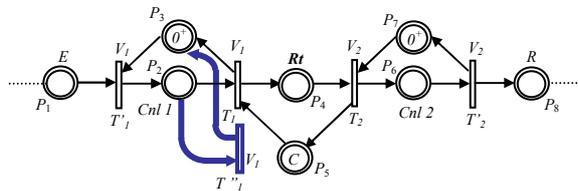


Figure 5. Modélisation d'une ligne de transmission avec pertes.

– Modélisation d'une ligne de communication dans un environnement extérieur : La ligne décrite par la Figure 5 ne tient pas compte de l'environnement extérieur. Cependant, une ligne de communication partage ses ressources, et notamment son buffer, avec d'autres ressources extérieures. On peut alors choisir de considérer le cas d'un seul émetteur extérieur connecté, ou celui de deux émetteurs extérieurs ou trois, sauf que ce nombre n'est jamais connu à l'avance. Il en est de même des instants de connexion et des débits respectifs de ces émetteurs extérieurs. Afin de considérer le cas général où nous n'avons aucune donnée précise sur l'environnement où baigne la ligne utilisée pour notre communication, nous considérons une seule entrée vers le routeur Rt qui représente la somme de toutes les entrées possibles. Cette entrée aura donc une allure aléatoire à l'image de ce qui peut se produire sur Internet (Figure 6).

Les places Rt_C et Rt_E de la figure 6 permettent de différencier le flux d'émission (issues de l'émetteur E) du flux extérieur (issues des émetteurs extérieurs E_{ext}) respectivement.

Selon que l'on considère le flux d'émission prioritaire au flux extérieur ou inversement, nous considérerons la transition T_1 prioritaire à T_3 ou inversement. Ainsi, le modèle gère aussi d'éventuels flux prioritaires sur une ligne.

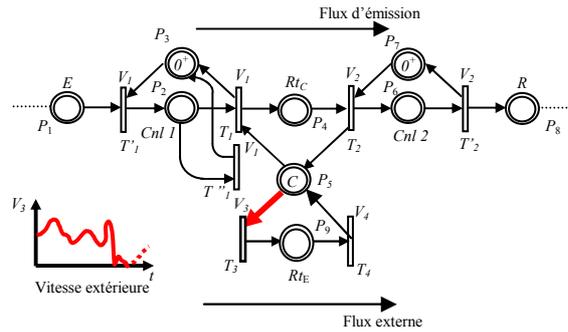


Figure 6. Modèle de la ligne partagée avec les flux extérieurs.

– Dynamique de la ligne de communication Nous allons considérer qu'en moyenne, les flux d'entrées du routeur sont supérieurs aux flux de sortie (dans le cas contraire, il n'y a ni remplissage du buffer ni pertes sur la ligne). Deux cas sont possibles : soit il reste de la place dans le routeur ($m_5 \neq 0$) et les messages qui arrivent peuvent être stockés, soit le routeur est plein et il y a alors conflit entre les transitions T_1 et T_3 .

- Tant que $m_5 \neq 0$, chacune des transitions T_1 , T_2 , T_3 et T_4 est fortement validée. Toutes les vitesses instantanées de franchissement associées à ces transitions sont égales à leurs vitesses maximales respectives. Les dynamiques des places P_4 , P_5 et P_9 s'écrivent sous la forme :

$$\dot{m}_4 = V_1 - V_2, \quad (5)$$

$$\dot{m}_5 = (V_2 + V_4) - (V_1 + V_3), \quad (6)$$

$$\dot{m}_9 = V_3 - V_4, \quad (7)$$

L'évolution de marquages de ces places s'écrit:

$$m_4(t) = (V_1 - V_2)t, \quad (8)$$

$$m_5(t) = C + [(V_2 + V_4) - (V_1 + V_3)]t, \quad (9)$$

$$m_9(t) = (V_3 - V_4)t, \quad (10)$$

Avec: $m_4(t) + m_5(t) + m_9(t) = C$ toujours vérifié.

Les dynamiques des places relatives au canal 2, P_6 et P_7 sont nulles et les marquages associés à ces places restent constants. L'équation de vitesse qui régit le canal 1 s'écrit sous la forme: $v_1'(t) = v_1(t) + v_1''(t)$. La vitesse $v_1'(t)$ et $v_1(t)$ étant égales à V_1 , $v_1''(t) = 0$, i.e., il n'y a pas de pertes sur la ligne tant qu'il y a suffisamment de place dans le buffer.

- Si $m_5 = 0$, T_1 et T_3 sont faiblement validées et il y a conflit entre ces deux transitions. Nous calculons les expressions de v_1 et v_3 selon la méthode de résolution des conflits donnés dans (David *et al.*, 2005):

$$v_1 \leq \min(V_1, v_1', v_2 + v_4 - v_3),$$

$$v_3 \leq \min(V_3, v_2 + v_4 - v_1),$$

$$v_1 + v_3 = v_2 + v_4.$$

T_3 étant prioritaire à T_1 , on a $v_3 = \min(V_3, v_2 + v_4)$ et v_1 est calculé en fonction de la valeur de v_3 . Ainsi, si $v_3 = V_3$, $v_1 = (v_2 + v_4) - V_3$ et $v_1' = V_1 + v_1$ et si $v_3 = v_2 + v_4$ alors $v_1 = 0$ et $v_1' = V_1$. En d'autres termes, le débit extérieur étant prioritaire, il prend toute la place que lui permet sa vitesse extérieure V_3 et le flux d'émission occupe la place restante dans le buffer. La vitesse des pertes varie selon celle des précédentes vitesses.

- Graphe d'évolution du RdP continu Les dynamiques citées ci-dessus peuvent être représentées par le graphe d'évolution de la ligne. Cependant, nous ne pouvons pas prévoir les évolutions des marquages en considérant le caractère aléatoire de la vitesse extérieure v_3 . Supposons que v_3 est constante (en considérant, par exemple, le système sur un intervalle de temps assez petit pour pouvoir assimiler v_3 à une valeur fixe) et prenons les cas où $V_3 > V_4$ puis celui où $V_3 < V_4$ (dans les deux cas, $V_1 > V_2$).

- $V_3 > V_4$:

Le graphe d'évolution correspond à celui de la figure 7.a. Le vecteur de marquage correspond aux places m_2 , m_3 , m_4 , m_5 et m_9 respectivement et le vecteur des vitesses correspond aux vitesses v_1 , v_1' , v_1'' , v_2 , v_3 et v_4 respectivement. Les deux branches correspondent aux deux cas possibles où $v_3 = V_3$ et où $v_3 = V_2 + V_4$ lorsque $m_5 = 0$. Dans les deux cas, on obtient : $\dot{m}_4 = V_4 - V_3 < 0$ et $\dot{m}_9 = V_3 - V_4 > 0$. Ainsi, la place m_4 se vide tandis que m_9 se remplit. Lorsque $m_4 = 0$ (à t_2 par l'IB2 et à t_3 par IB3), les dynamiques se stabilisent aux vitesses données par l'IB4.

- $V_3 < V_4$:

Ce cas correspond à celui où seule la ligne émettrice charge le buffer (car $V_1 > V_2$). Le graphe d'évolution correspond à celui de la figure 7.b. Tant que $m_5 \neq 0$ (IB1), les vitesses sont maximales excepté le débit de sortie extérieur qui s'aligne sur le débit d'entrée ($v_4 = V_3$). Les dynamiques des places du routeur sont:

$$\begin{aligned} \dot{m}_9 &= 0, \\ \dot{m}_4 &= V_1 - V_2 > 0, \\ \dot{m}_5 &= V_2 - V_1 < 0 \end{aligned}$$

Ainsi, le marquage de m_5 diminue tandis que celui de m_4 augmente. Lorsque $m_5 = 0$ (IB2), $v_3 = \min[V_3, v_2 + V_4] = V_3$, $v_1 = \min[V_1, v_2 + v_4 - v_3] = \min[V_1, V_2] = V_2$ (car $v_4 = v_3 = V_3$ et $V_2 < V_1$) et $v''_1 = V_1 - V_2$.

Il apparaît dans cet exemple que malgré la priorité du flux extérieur sur le flux l'émission, le buffer se charge des données d'émission et non du flux extérieur.

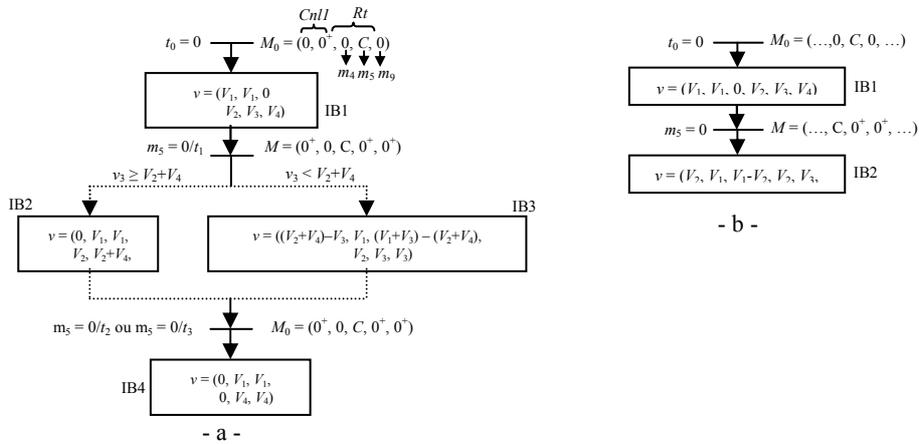


Figure 9. a) Graphe d'évolution de la ligne de communication avec $V_3 > V_4$. b) Graphe d'évolution de la ligne de communication avec $V_3 < V_4$.

4. 2. Le modèle RdP discret d'une ligne de communication TCP/IP

Les deux informations pertinentes sur lesquelles se basent les protocoles TCP sont l'arrivée ou non d'acquiescement et l'expiration du temporisateur de retransmission. Ces deux informations servent à estimer l'état du réseau pour moduler au mieux la taille des données envoyées.

Les données sont envoyées en continu sur la ligne, i.e., dès qu'une fenêtre est finie d'être envoyée on envoie immédiatement la fenêtre suivante. Ceci implique la présence de plusieurs fenêtres simultanément sur la ligne avec leurs acquiescements et leurs temporisations propres. Nous allons voir comment un modèle coloré peut gérer cette superposition de fenêtre sur une ligne.

4.2.1. Modélisation de l'algorithme slow start

Durant cette partie de la transmission, chaque acquiescement reçu augmente la nouvelle fenêtre de congestion de deux nouveaux segments.

Observons la figure 8. Elle représente le modèle (que l'on va compléter plus loin) d'une seule fenêtre lancée sur la ligne. Une fenêtre est activée (place *Autss1*

marquée) et sa temporisation associée est lancée (place *Tempo1* marquée) par le franchissement de T_1 . Un acquittement est envoyé par le récepteur à chaque fois qu'il reçoit un segment. S'il y a l'équivalent d'un segment (m) dans le récepteur R , un acquittement est envoyé. Au niveau du modèle, cela se traduit par le franchissement de T_3 lorsque $R = m$. Une quantité d'information continue, ici m , se transforme en nombre discret, ici nombre d'acquittements, par le biais de la D-transition T_3 . La place *ack1* représente ainsi le nombre d'acquittements reçus par l'émetteur.

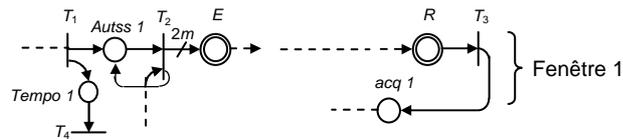


Figure 8. Envoie d'une seule fenêtre de donnée sur la ligne de transmission.

Nous allons considérer maintenant le cas où plusieurs fenêtres se chevauchent sur la ligne de communication. La figure 9 représente le cas où 3 fenêtres au plus peuvent se chevaucher. Ainsi, trois parties identiques se répètent où chaque partie (ou étage) représente l'envoi d'une fenêtre. La ligne continue est représentée en pointillé (mis à part E et R) pour ne pas encombrer le modèle. Supposons que l'on active et lance la première fenêtre (les places *Tempo1* et *Autss1* marquées). Les acquittements de cette première fenêtre déterminent la taille de la seconde fenêtre. Remarquons que pour chaque acquittement arrivé, deux segments sont créés dans la seconde fenêtre pour assurer l'évolution exponentielle des données. La seconde fenêtre est activée puis lancée lorsque la première fenêtre est complètement envoyée. De même, les acquittements de cette seconde fenêtre vont déterminer la taille de la troisième fenêtre que je vais activer et lancer à son tour lorsque j'aurais fini d'envoyer la seconde fenêtre. Si l'on considère qu'il n'y a que trois fenêtres au plus qui se chevauchent, nous pouvons boucler le schéma (arc *acq3* vers T_1) à la fin de la 3^{ème} fenêtre. D'une part, nous ne savons pas à l'avance le nombre de fenêtres qui vont se chevaucher. D'autres part, s'il y a plusieurs fenêtres qui se chevauchent, le modèle risque d'être très gros. La 2^{ème} solution proposée alors est de considérer un modèle coloré avec l'ensemble des couleurs donné par les différentes fenêtres présentes simultanément sur la ligne.

Commençons pas donner une petite définition de ce que sont les RdP colorés. Les RdP colorés sont très pratiques lorsqu'il s'agit d'un système qui possède plusieurs composants aux comportements identiques (comme c'est notre cas ici). La représentation de tels systèmes par des RdP ordinaires peut donner des modèles de taille inexploitable. Il s'agit alors, par RdP colorés, d'associer une *couleur* à chaque partie et d'effectuer ensuite un *pliage* de ces différentes parties. Une couleur est une façon de différencier des jetons lorsqu'on a besoin d'informations détaillées sur un marquage.

Le modèle RdP hybride coloré d'une ligne de transmission est donné par la figure 10. Une couleur est associée à chaque fenêtre de données envoyée. Ainsi, les places contiennent des marquages associés à des numéros de fenêtres. Un marquage $\langle m$,

f_i signifie que cette place contient la quantité m de donnée correspondant à la fenêtre d'envoi f_i . L'ensemble des couleurs associées aussi bien aux places qu'aux transitions est nommée $F=(f_1, f_2, \dots, f_n)$, n étant le nombre maximal représentant le nombre de fenêtres présentes sur la ligne.

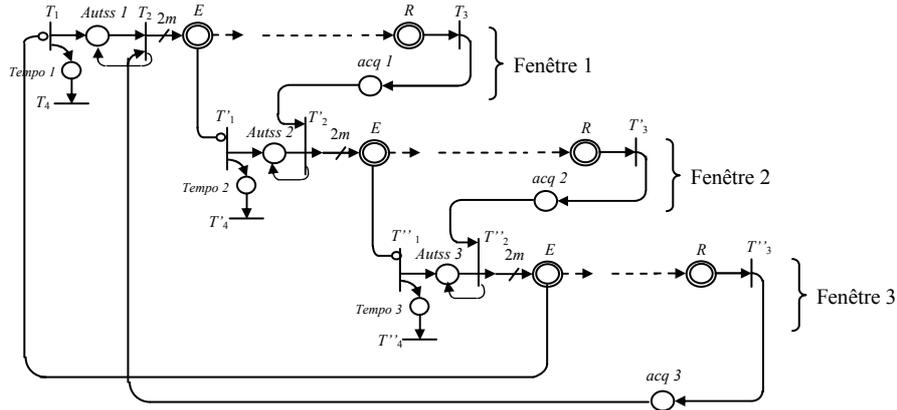


Figure 9. Modèle non coloré du chevauchement de trois fenêtres de transmission

Au niveau des transitions T_5 et T_7 , on voit donc l'interaction directe, d'une part entre le flux continu sur la ligne qui se transforme en information discrète pour le protocole et d'autre part, ces mêmes informations discrètes (nombre d'acquitements) servant à gérer la quantité de flux sur la ligne. Ce modèle montre les parties continues et discrètes sur une ligne de communication et leur constante interaction formant ainsi un modèle hybride.

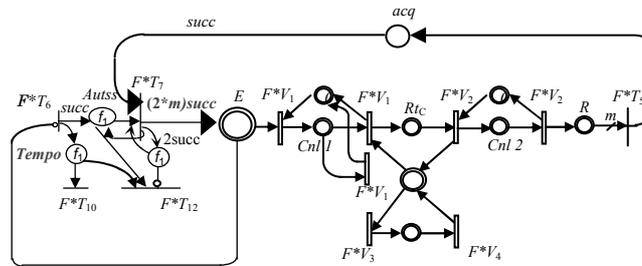


Figure 10. Modèle coloré d'une ligne soumise au protocole slow start.

4.2.2. Modélisation de l'algorithme congestion avoidance

En congestion avoidance, pour chaque *ack* arrivé, on ajoute un segment dans la nouvelle fenêtre. De même que pour le protocole slow start, l'évolution linéaire va nécessiter l'utilisation des RdP colorés pour gérer le chevauchement des données.

par seconde), $V_2=1000\text{Koct/s}$. La vitesse V_3 est aléatoire et varie entre 800Koct/s et 2800Koct/s (figure 13.b). $V_4=1500\text{Koct/s}$, $C=600\text{Koct}$, $\delta=10\text{ms}$. La valeur des débits est prise de sorte que les débits d'entrées du routeur soient, en moyenne, supérieures aux débits de sorties du routeur (charge du routeur).

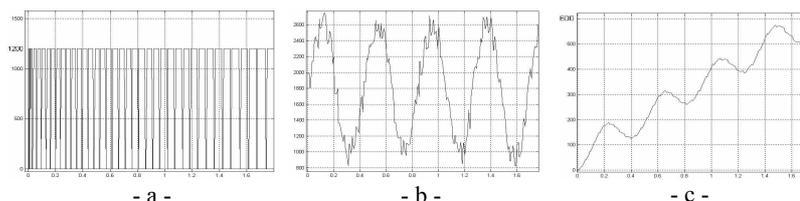


Figure 13. a) Vitesse d'entrée de l'émetteur . b) Vitesse d'entrée extérieure. c) Dynamique d'évolution routeur.

- Tant que le buffer n'est pas plein: (figure 13), le buffer accepte les messages qui proviennent de la source E et des sources extérieures, les vitesses de franchissement du flux d'émission (v_{11}) et du flux extérieur (v_3) sont maximales (figures 13.a and 13.b) et le buffer se remplit selon la courbe de la figure 13.c où les oscillations observées sont dues à la vitesse extérieure.

- Lorsque la charge du buffer atteint sa capacité maximale C , certaines données d'émissions et extérieures commencent à se perdre. La figure 14.a indique l'état du buffer après le début des pertes. Périodiquement, le niveau du buffer se maintient à C puis se vide légèrement avant d'atteindre C à nouveau. Si l'on observe les moments où le buffer se remplit et se vide, l'on voit que ces moments correspondent à ceux où les débits extérieurs augmentent puis diminuent (v_3 , Figure 14.c). Si toutes les vitesses sont constantes, le système se stabilise dès que le buffer est plein (cas des dynamiques vues dans les graphes d'évolution des Figures 7.a et 7.b). Les fluctuations de la vitesse extérieure font que la charge du buffer fluctue aussi. Réellement, les charge et décharge du buffer ne font plus que suivre les fluctuations du débit extérieur.

Les vitesses d'émission (Figures 14.b) et extérieure (Figures 14.d) baissent à leur tour lorsque le buffer est plein. Ainsi, on voit sur ces deux figures qu'aux périodes où le buffer se vide un peu, ces vitesses redeviennent maximales, par contre lorsque le buffer est plein, la vitesse d'émission va jusqu'à zéro tandis que la vitesse extérieure ne fait que diminuer (elle baisse jusqu'à la valeur V_4). Ceci est évidemment dû à la priorité des flux extérieurs sur le flux d'émission. Ainsi, la vitesse extérieure s'aligne aux débits de sortie du buffer et la vitesse d'émission s'annule.\

La figure 15 montre l'évolution de la taille des fenêtres de transmission. Tant qu'il n'y a pas de pertes sur la ligne, la taille de la fenêtre évolue d'abord de façon exponentielle (agrandissement sur la Figure 12.b), puis de façon linéaire (agrandissement sur la Figure 12.c). Au moment où les pertes commencent, le système est réinitialisé et la fenêtre est remise à un (un segment), sa valeur initiale. La transmission reprend à slow start (Figure 12.d). Après le début des pertes, il y a

des périodes de reprise de transmission et d'autres où la fenêtre ne dépasse pas la taille d'un segment. Ces deux phases correspondent à celles de pertes et de reprises successives dues aux variations de v_3 , si la valeur de v_3 baisse, la retransmission reprend, sinon, la transmission est très faible (de l'ordre d'un segment par *Tempo*).

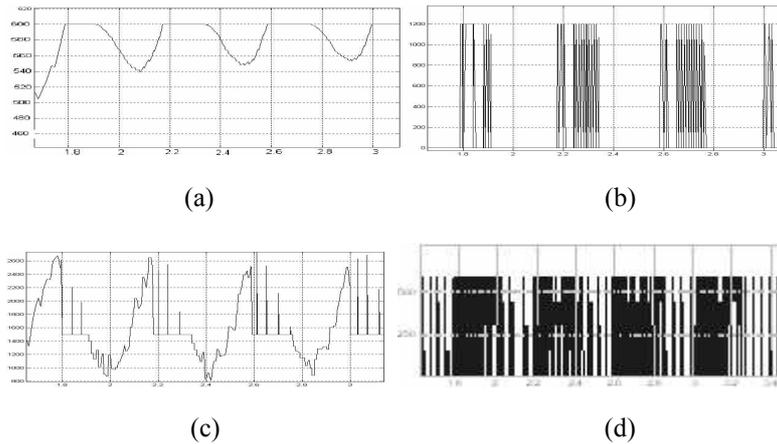


Figure 19. Evolution of: a) emission speed, b) external speed, c) loss speed in accordance with buffer state.

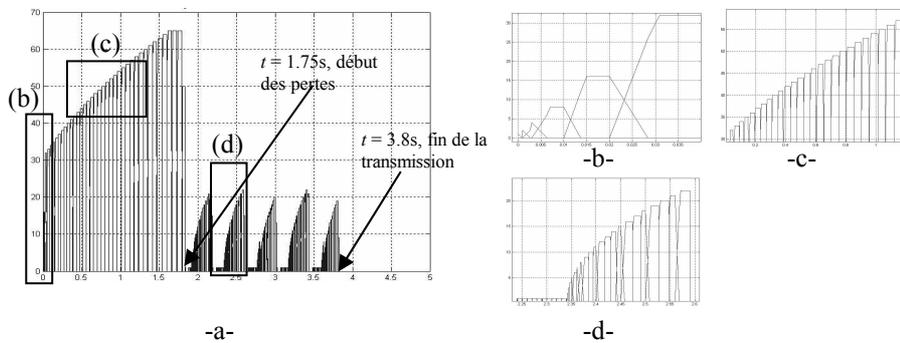


Figure 12: Congestion window evolution during a) the entire transmission, b) slow start, c) congestion avoidance, d) loss recovery.

5.1 Indicateurs de performance

- Temps de transmission : Le temps de transmission global est directement lisible dans les simulations. En effet, c'est l'instant où la vitesse à la réception et la fenêtre de congestion sont nulles. La figure 16.a représente le délai de transmission de la quantité totale de données $U=3\text{Moct}$. Ce paramètre est mesuré plusieurs fois en

faisant varier la taille du buffer C ; tous les autres paramètres restent constants. Lorsque la taille du buffer est grande, les pertes se produisent de façon tardive car le buffer met du temps pour se remplir. Ainsi, peu de pertes se produisent et le système perd moins de temps dans les retransmissions. Lorsqu'au contraire, C est faible, beaucoup de pertes se produisent et beaucoup de données doivent être retransmises, les délais de transmission se voient retardés. En conclusion, les délais de transmission sont faibles lorsque C augmente.

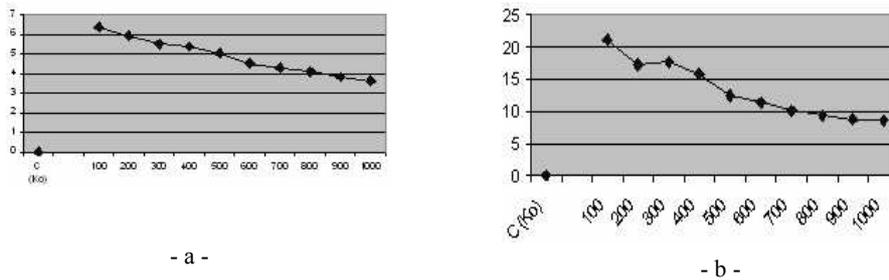


Figure 15. a) Délais de transmission en fonction de la capacité du buffer, b) Taux de pertes en fonction de la capacité du buffer.

- Le taux de pertes : Le taux de pertes représente le rapport entre la quantité de pertes (vues par les protocoles) sur la quantité de données totale envoyée. Les pertes vues par les protocoles sont celles détectées après expiration des temporisateurs. En faisant varier le paramètre C , ce taux de pertes nous donne une appréciation de la quantité de données que le système a du perdre et donc retransmettre.

On voit sur La figure 16.b que ce taux augmente au fur et à mesure que la capacité C du routeur diminue. Ce résultat rejoint celui relatif aux délais de transmission et les deux sont liés car les délais dépendent aussi de la quantité de pertes sur la ligne.

- Comparaison entre les pertes vues par les protocoles et celles vues par le modèle : Nous avons décrit précédemment les pertes vues par les protocoles comme étant celles détectées suite à l'expiration des temporisateurs. Du point de vue modèle, la transition T''_1 (Figure 12) représente la quantité de données réelles qui n'ont pas trouvés place dans le routeur, i.e., les pertes réelles. Une comparaison pertes réelles / pertes vues par les protocoles est effectuée et donnée par la figure 17. La première observation est que la courbe qui représente les pertes vues les protocoles est constamment au dessus de la courbe des pertes réelles. Ce résultat est prévisible. D'autre part, en faisant varier la capacité du routeur C , La différence entre les deux courbes croît. En effet, les pertes estimées par les protocoles comprennent aussi les données encore présentes dans le buffer. Or ces dernières sont d'autant plus nombreuses que la taille du buffer est grande. Cette comparaison pertes réelles / pertes vues par les protocoles peut aussi servir de point d'évaluation des performances des protocoles mêmes. En effet, plus les protocoles sont bien adaptés à l'état du réseau, plus les deux courbes (pertes réelles, pertes protocoles) seront proches.

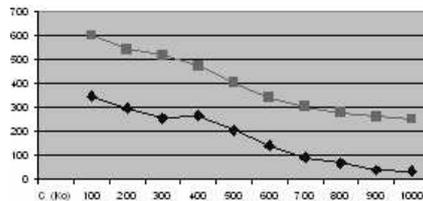


Figure 17: A comparison between losses found by TCP protocol and real losses.

6. Conclusions et perspectives :

Nous avons traité tout au long de ce papier de la modélisation de phénomènes complexes et de différentes natures présents dans les réseaux.

Le réseaux Internet est une infrastructure de niveau mondiale requiert l'intégration d'une organisation pointue et hiérarchique, d'un contrôle puissant et d'études de performances et de qualité de service à tous les niveaux des réseaux. Pour notre part, nous nous sommes penchés sur les réseaux TCP/IP et plus particulièrement sur le contrôle des pertes et l'évaluation de performances d'une ligne de transmission. Avant d'entamer un tel travail, nous avons d'abord modélisé le système considéré.

Deux parties distinctes apparaissent dans la modélisation d'une telle ligne. D'une part, les données (succession de bits) sont transmises à des débits généralement de l'ordre du mega bits / secondes. Ce rapport débit/taille de données nous a fait considérer qu'un flux continu peut représenter une très bonne approximation du passage discret de bits (ou d'octets). D'autre part, la transmission est régie par les protocoles de contrôle de flux du TCP/IP qui sont constitués d'un ensemble de consignes discrètes (arrivées d'acquittements, expiration de temporisateurs, etc.). Ce modèle comprend donc des dynamiques continues de par le passage de flux de messages sur les lignes et des dynamiques discrètes de par les consignes discrètes des protocoles de contrôle. Ces deux parties interagissant entre elles pour donner une dynamique continue ponctuellement modifiée par une dynamique discrète. Les dynamiques des réseaux sont donc hybrides.

Nous avons pu, grâce aux RdP hybrides mettre en évidence des phénomènes complexe tels que la charge des buffers, la baisse des débits à l'entrée des routeurs, la séparation des flux, la gestion des priorités ou encore les pertes. Afin d'évaluer les performances de la ligne et d'étudier son comportement face aux changements des conditions de transmission, un large panel de simulations a été effectué (Bitam *et al.*, 2005). Nous avons pu visualiser des performances telles que les débits, les délais de transmission, les taux de pertes. Les différents paramètres qui gèrent l'évolution d'une transmission tels que les temps de boucle et les temporisateurs ont aussi été visualisés. Une comparaison entre les pertes réelles produites sur la ligne et celles calculées par les protocoles est possible grâce au modèle RdP hybride qui nous donne la quantité exacte de données perdues. Cette comparaison constitue un bon indice d'évaluation de performances des protocoles.

Dans les routeurs tels que modélisés dans ce présent travail, la présence de flux différents intervient en terme de partage de la capacité totale du routeur. Une modélisation plus précise des routeurs doit tenir compte de la gestion de file d'attente des données en sortie. Un premier travail de prise en compte de file d'attente FIFO dans les routeurs a été effectué avec comme objectif d'intégrer les délais supplémentaires générés par la file dans le modèle RdP. Une comparaison entre les résultats obtenus avec ou sans prise en compte de la file d'attente permet d'arriver à une connaissance a priori des taux d'occupation des buffers sur une ligne de transmission. Une formalisation de ces travaux constitue une perspective immédiate de la continuité de ces travaux de recherche.

7. Bibliographie :

- Alla H., David R., « On Hybrid Petri Nets », *Discrete Event Dynamic Systems: Theory and Applications*, vol. 11, 2001, n° 9-40.
- Alla H., David R., « A modeling and analysis tool for discrete events systems: continuous Petri net », *Performance Evaluation*, vol. 33, 1998, n° 175-199.
- Bitam M., Alla H., « Performance Evaluation of a TCP/IP Transmission Using Hybrid Petri Nets », *IEEE International Conference on Computer Systems and Applications, AICCSA*, Cairo, Egypt, 3-6 Janvier 2005.
- Bitam M., « Modélisation et analyse des protocoles de communication TCP/IP », *Conférence Internationale Francophone d'Automatique CIFA'04*, Douz, Tunisie, 22-24 Novembre 2004.
- Bitam M., Alla H., « Modeling of a Communication Network under TCP/IP protocols using Hybrid Petri nets », *IFAC Conference on Analysis and Design of Hybrid Systems, ADHS03*, Saint Malo, France, 16-18 Juin 2003.
- David R., Alla H., *Discrete, Continuous, and Hybrid Petri Nets*, Springer-Verlag Berlin Heidelberg, 2005.
- Dubois D., Forestier J.-P., « Productivité et en-cours moyens d'un ensemble de deux machines séparées par un stock », *RAIRO Automatique*, vol. 16, n°6, 1982, p. 105-132.
- Gershwin S.B., Schick I.C., « Continuous Model of an Unreliable Two-Stage Material Flow System With a Finite Interstage Buffer », Technical report MIT LIDS-R-1032, Septembre, 1980.
- Jacobson V., « Congestion Avoidance and Control ». *Proceedings of SIGCOMM '88*, August 1988, pp. 314-329.
- Mézghani M., « Modélisation et évaluation de performances d'un réseau de distribution de la ville d'Aix-Les-Bains », Student report, Laboratoire d'Automatique, INP Grenoble, juin 1994.
- Pujolle G., *Les réseaux*, Aux éditions Eyrolles, 2005.
- Tanenbaum V., *Networks*. Prentice Hall, 1997.
- Zaytoon J., *Systèmes dynamiques hybrides*, Paris, Hermès science Pub., 2001.