

UNIVERSITÉ PAUL SABATIER

Réseaux Temps Réel

---

- Bureau d'étude -  
NETWORK CALCULUS APPLIQUÉ AU RÉSEAU  
AFDX

---

*Auteurs :*

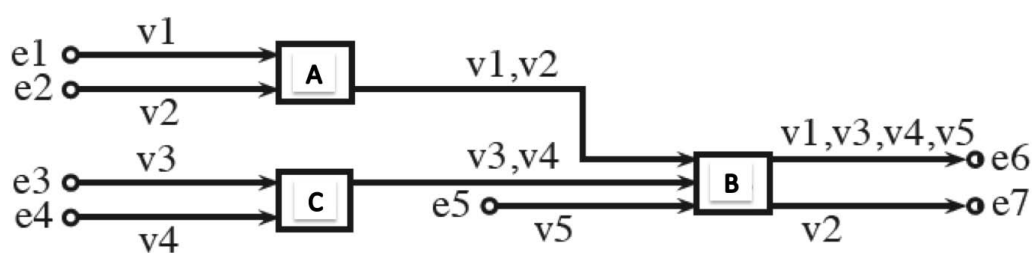
Lucien RAKOTOMALALA

David TOCAVEN

*Encadrant :*

Euriell LE CORRONC

Michaël LAUER





# Table des matières

<b>Introduction</b>	<b>1</b>
<b>1 Étude du n[Pleaseinsertintopreamble]ud A</b>	<b>2</b>
1.1 Courbes d'arrivée $\alpha$ . . . . .	2
1.2 Courbe de service $\beta$ . . . . .	3
1.3 Courbe d'arrivé du nœud A . . . . .	4
1.4 Délai et backlog pire cas . . . . .	4
1.5 Courbes d'arrivé du port 1 et 2 de $B$ . . . . .	4
<b>2 Etude du réseau complet</b>	<b>5</b>
2.1 Analyse générale du réseau . . . . .	5
2.1.1 Courbes d'arrivées de B et C . . . . .	5
2.1.2 Courbes de service de B et C . . . . .	5
2.1.3 Étude de l'ensemble du réseau . . . . .	5
2.2 Borne sur les pires temps de traversée et sur le pire backlog . . . . .	5
2.2.1 Pire délai de traversée de bout-en bout . . . . .	5
2.2.2 Pire backlog du réseau . . . . .	5
2.3 Amélioration du pire délai de traversée de bout-en bout . . . . .	5
2.3.1 Dépendance des flux et tracé des nouvelles courbes . . . . .	5
2.3.2 Re-calcul des délais pire cas et conclusion . . . . .	5
<b>3</b>	<b>6</b>
<b>4 Conclusion</b>	<b>7</b>
<b>Annexes</b>	<b>9</b>
<b>TITRE</b>	<b>9</b>
TITRE . . . . .	9
<b>Annexe 2 - TITRE</b>	<b>10</b>

# Introduction

# Chapitre 1

## Étude du nœud A

Dans cette partie, nous étudierons un premier bloc du réseau. Nous calculerons les courbes d'arrivée du nœud et la courbe de service pour comprendre sa courbe de sortie et ainsi mesurer sa portée sur le reste du réseau.

Les tracés de l'ensemble des courbes ont été obtenus par l'interpréteur en ligne <http://realtimeatwork.com/minplus-playground>, il s'agit d'un interpréteur qui permet entre autre le tracé de courbes affines mais aussi le calcul en algèbre (min,+).

Nous avons décidé de d'utiliser et de fixer les unités sur lesquelles nous baserons nos courbes. Nous avons choisi de travailler en **octets** (axe des ordonnées) et le temps (axe des abscisses) sera en **millisecondes**. Nous verrons en synthèse à ce rapport si nos choix pour ces unités étaient judicieux.

### 1.1 Courbes d'arrivée $\alpha$

Pour déterminer la courbe d'arrivée, nous allons tout d'abord analyser les données que nous pouvons recueillir sur le bloc A. Nous relevons deux flux d'entrée  $v1$  et  $v2$ . Pour connaître la courbe d'arrivée  $\alpha^A$ , nous devons tracer les deux courbes d'entrée des deux flux entrants, courbes que nous affichons en figure (??)

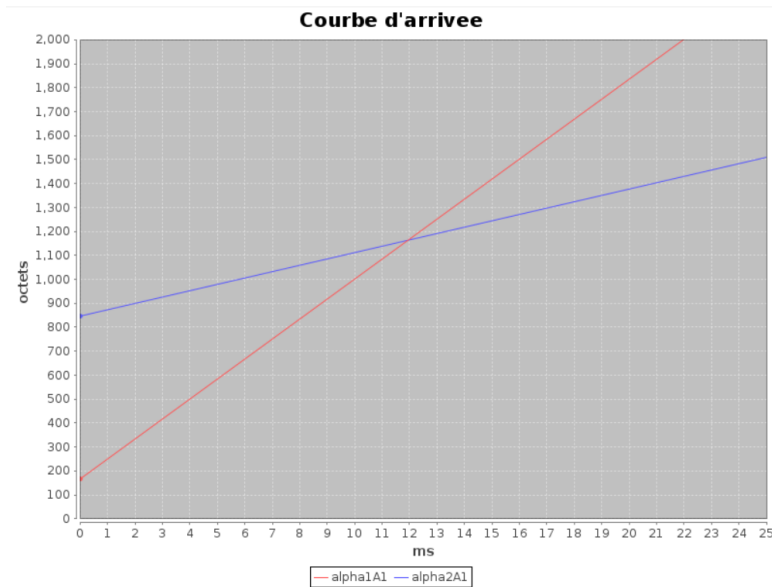


FIGURE 1.1 – Courbe d'arrivée  $\alpha$  du flux  $v1$  (rouge) et du flux  $v2$  (bleu)

Ces courbes ont été obtenues à l'aide des informations sur les *BAG* et les  $s_{max}$  de  $v1$  et  $v2$ . Pour obtenir des données correspondantes aux unités choisies en 1, nous devons utiliser la relation suivante qui lie la taille maximale d'une trame  $L_j^{max}$  et la charge utile maximale  $s_j^{max}$  d'un *Virtual Link* (VL)  $j$  :

$$L_j^{max} = \max(s_j^{max}, 17) + 47 \quad (1.1)$$

Nous observons donc avec cette équation que la taille maximale d'une trame est strictement supérieure à 50. À l'aide de cette équation, nous sommes capables d'établir la pente  $a_j$  de la courbe des données maximales ainsi

que son offset  $b_j$  de décalage qui peuvent arrivées dans  $A$  avec

$$a_j = \frac{L_j^{max}}{BAG_j} \quad (1.2)$$

$$b_j = L_j^{max} \quad (1.3)$$

Dans notre cas, nous obtenons l'application numérique suivante :

$$a_1 = \frac{L_1^{max}}{BAG_1} = \frac{\max(s_j^{max}, 17) + 47}{BAG_1} = \frac{167}{2} = 83.5$$

$$b_1 = \max(s_j^{max}, 17) + 47 = 167$$

Avec l'interpréteur, nous pouvons obtenir la courbe affine avec la commande :

```
alpha1A1 := affine(83.5, 167) //echelle octets/ms
```

De même pour le flux  $v_2$ , nous obtenons comme application numérique :

$$a_2 = 26.46$$

$$b_2 = 847$$

La courbe obtenu est de la forme sceau percé, nous avons une quantité maximale émise instantanément par le bloc  $A$  et un débit moyen maximal.

## 1.2 Courbe de service $\beta$

Nous allons maintenant établir la courbe de service disponible par le nœud  $A$ . Cette fonction du port desortie de  $A$  est établit en fonction de la latence technologique  $\mu$  déterminé et du débit du port de sortie. Notre étude est calibré selon lesuniés défini en1, il est donc important de respecter ces conditions dès maintenant pour éviter tout problème fuur d'unité. Nous relevons :

$$debit = 100Mb/s = 12500 \text{ Octets}/ms \quad (1.4)$$

$$\mu = 16\mu s = 0.016ms \quad (1.5)$$

Pour établir cette fonction sur *Network Calculus*, nous avons choisi d'établir une fonction affine qui admet comme pente le débit du port de sortie mais qui subit un décalage, i.e un *delay* dans l'interpréteur, pour modéliser la latence technologique. La courbe en ?? est obtenu avec la ligne de commande :

```
betaA := affine(12500,0) * delay(0.016)
```



FIGURE 1.2 – Courbe de service du nœud  $A$

Cette courbe est du type latence-taux, car notre modèle admet une latence technologique et dispose d'un débit de sortie. Pour résumé, le modèle du nœud  $A$  reçoit des données dans une courbe d'arrivé de type sceau percé pour les traiter avec une courbe de service du type latence-tau. Selon nos estimations, le nœud va donc lisser le débit des flux  $v_1$  et  $v_2$ .

### 1.3 Courbe d'arrivé du nœud A

Nous nous intéressons maintenant au cumul des flux dans l'entrée du nœud  $A$ . Pour cela, nous allons utiliser le résultat expliqué en pendant les travaux dirigés qui expriment :

$$\alpha^A = \alpha_1^A + \alpha_2^A \quad (1.6)$$

soit la somme des pentes  $r_i$  et des  $L_i^{max}$  de  $\alpha_1^A$  et  $\alpha_2^A$ . Nous obtenons par le calcul une fonction affine décrit dans l'équation que nous comparons avec les courbes  $\alpha_1^A$  et  $\alpha_2^A$  dans la figure ??.

$$\alpha^A = \gamma_{r,b} \text{ avec } r \text{ la pente et } b \text{ les tailles maximales de la trame } L_i \quad (1.7)$$

$$r = r_1 + r_2 = a_1 + a_2 \text{ et} \quad (1.8)$$

$$b = b_1 + b_2 = b_1 + b_2 \quad (1.9)$$

$$(1.10)$$

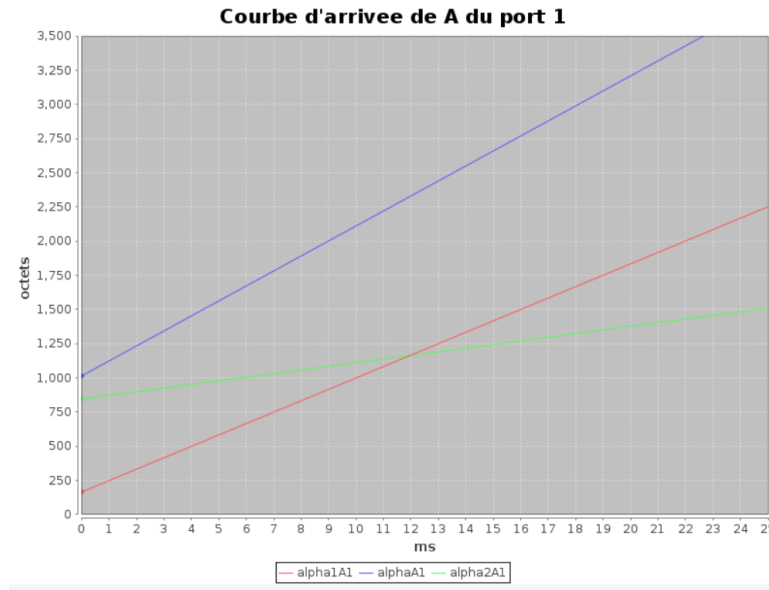


FIGURE 1.3 – Courbe de d'arrivé du nœud A

### 1.4 Délai et backlog pire cas

Le délai pire cas et le backlog permettent de se concentrer sur l'étude des temps de traversé du système. Nous en aurons besoin dans le suite de ce rapport, c'est pourquoi nous commençons l'étude dans cette partie.

Pour déterminer ce temps et cette quantité de donnée, nous utilisons un résultat donné en cours qui explique les relations suivantes : le délai pire cas est la plus grande différence horizontale entre la courbe d'arrivé et la courbe de service d'un nœud noté  $hDev$ . Respectivement, le backlog pire cas est la plus grande différence verticale entre les deux courbes noté  $vDev$ .

Nous obtenons ainsi un délai et une quantité qui représentent les pire cas possibles de traitement du nœud. Avec l'interpréteur de commande, nous pouvons calculer ces deux informations et nous obtenons :

$$hDev(\alpha^A, \beta^A) = 97.12ms \quad (1.11)$$

$$vDev(\alpha^A, \beta^A) = 1015 \text{ octets} \quad (1.12)$$

Nous ne connaissons pas les limites du système qui utilise ce réseau, nous ne pouvons donc pas établir de conclusions sur ces résultat. Nous pourrions les comparer une fois que les délai et backlog pire cas des autres nœud auront été calculé.

### 1.5 Courbes d'arrivé du port 1 et 2 de B

## Chapitre 2

# Etude du réseau complet

Dans ce chapitre, nous allons étendre l'étude précédente à l'ensemble du réseau. Nous allons commencer par une étude générale du réseau puis, nous étudierons les pire temps de traversé (WCTT) et les pires différences des volumes de données du réseau (backlog). Pour fini, nous verrons comment améliorer le WCTT et le backlog du réseaux.

### 2.1 Analyse générale du réseau

Dans un premier temps, nous allons calculer les courbes d'arrivées des flux  $v_3$ ,  $v_4$  et  $v_5$  dans B et C puis les courbes de service des ports de sorties de B et C.

#### 2.1.1 Courbes d'arrivées de B et C

#### 2.1.2 Courbes de service de B et C

#### 2.1.3 Étude de l'ensemble du réseau

### 2.2 Borne sur les pires temps de traversée et sur le pire backlog

#### 2.2.1 Pire délai de traversée de bout-en bout

#### 2.2.2 Pire backlog du réseau

### 2.3 Amélioration du pire délai de traversée de bout-en bout

#### 2.3.1 Dépendance des flux et tracé des nouvelles courbes

#### 2.3.2 Re-calcul des délais pire cas et conclusion



# Chapitre 3

Chapitre 4

Conclusion

# Annexes

## Annexe 1 - TITRE

## Annexe 2 - TITRE