

# Analyse de performance de tampons partagés par des tâches périodiques et indépendantes

J. Legrand, F. Singhoff, L. Nana, L. Marcé  
EA 3883, Université de Bretagne Occidentale  
20, av Le Gorgeu  
CS 93837, 29238 Brest Cedex 3  
{jlegrand,singhoff,nana,marce}@univ-brest.fr

## Résumé

*Cet article présente une approche pour l'étude de la faisabilité de systèmes temps réel répartis. Nous considérons qu'un système réparti est composé de sous-systèmes mono-processeur reliés entre eux par un réseau. Les sous-systèmes comprennent des tâches périodiques indépendantes et des tampons. Les tampons reçoivent des messages par l'intermédiaire du réseau ou d'autres tâches du sous-système. Les messages sont ensuite consommés par une tâche périodique. Nous proposons une approche pour tester la faisabilité de ces sous-systèmes. En particulier, nous proposons des critères maximums et moyens de performance pour l'étude des tampons. Ces critères sont basés sur la théorie des files d'attente et de résultats provenant des réseaux ATM. Des résultats de simulation permettent d'évaluer nos propositions.*

## Mots clefs

Ordonnancement temps réel, système réparti, tampon, file d'attente

# 1 Introduction

Le terme "système temps réel" regroupe l'ensemble des systèmes dont les spécifications contiennent des contraintes de nature temporelle. Dans les systèmes temps réel, il n'est pas suffisant de considérer uniquement l'exactitude des résultats produits par l'application temps réel, c'est à dire, de contrôler si la mise en œuvre des fonctionnalités correspond à leurs spécifications. La difficulté spécifique des systèmes temps réel est de s'assurer de leur faisabilité temporelle. Autrement dit, il faut vérifier le respect effectif de leurs contraintes temporelles.

Nous nous intéressons, dans cet article, à la faisabilité des systèmes temps réel répartis. Nous considérons qu'un système réparti est constitué de plusieurs processeurs reliés par un réseau. Ce modèle d'application correspond à une pratique industrielle que l'on retrouve, par exemple, dans l'avionique modulaire : le système est constitué de sous-systèmes communiquant par l'intermédiaire d'un réseau [Ari97]. Ces sous-systèmes sont éventuellement fournis par différents partenaires industriels. Le fonctionnement global du système est défini par un intégrateur de système. Nous supposons qu'un sous-système est composé de tâches **périodiques** et de **tampons**. Les tampons d'un sous-système collectent des informations issues du réseau ou produites par des tâches de ce même sous-système.

Nous proposons d'étendre les résultats de la théorie de l'ordonnancement temps réel à ce type d'architecture. Notre objectif est de préciser la notion de faisabilité pour le modèle d'application étudié. Cette faisabilité implique de vérifier non seulement le respect des contraintes temporelles des tâches, mais aussi de s'assurer **qu'il n'y a pas de situation de débordement de tampon**. La vérification du respect des contraintes temporelles des tâches est accomplie grâce aux méthodes classiques de faisabilité (borne sur le taux d'utilisation processeur, calcul du temps de réponse [JP86, ABRT93], ...). En ce qui concerne les tampons, la théorie des files d'attente propose des critères de performance pour les dimensionner. Les tâches qui consomment les messages sont alors soumises à une contrainte de précedence : elles sont activées dès qu'un message arrive dans le tampon.

Malheureusement, dans ce cas, la vérification du respect des contraintes temporelles des tâches est plus difficile. En effet, les tests de faisabilité disponibles pour des jeux de tâches avec précedence imposent des restrictions sur leurs paramètres. Il faut notamment que les périodes des tâches liées par une contrainte de précedence soient identiques [RCR01]. En outre, les tâches sont activées selon le type d'arrivée des messages. Peu de résultats concernant la faisabilité existent en dehors des modèles de tâches classiques tels que les modèles périodiques.

Nous explorons une voie différente qui consiste à privilégier les jeux de tâches pour lesquels les tests de faisabilité sont simples, peu contraignants et permettent d'obtenir des garanties déterministes. Nous considérons donc les jeux de tâches périodiques sans contrainte de précedence. La vérification du respect des contraintes temporelles est alors relativement aisée [Riv98]. Toutefois, la difficulté est reportée sur le dimensionnement des tampons. Il n'est plus possible d'appliquer directement les résultats de la théorie des files d'attente.

Nous proposons d'étudier les tampons des sous-systèmes mono-processeur selon l'approche suivante :

- Lorsque la durée minimum entre deux arrivées de messages est connue, des bornes maximum sur le taux d'occupation du tampon et le temps d'attente des messages permettent d'avoir des garanties sur le bon fonctionnement de l'application.
- Lorsque seule la durée moyenne entre deux arrivées de messages est connue, des critères moyens de performance, tels que le temps moyen d'attente et le taux moyen d'occupation, permettent d'évaluer les caractéristiques du tampon de manière à diminuer le risque de débordement.

Le plan de l'article est le suivant. Après une description succincte du modèle de tâche et de la théorie des files d'attente, nous exposons les caractéristiques d'une loi de service que nous proposons, la loi de service  $P$ . Nous déterminons ensuite les critères moyens et maximums de performance nécessaires à l'étude des tampons. Puis, nous présentons des résultats de simulations afin d'évaluer nos propositions. Finalement, nous concluons.

## 2 Modèle de tâche et théorie des files d'attente

Une tâche périodique modélise un traitement répétitif. Une tâche périodique  $i$  d'un sous-système est décrit par une période  $P_i$ , une capacité  $C_i$  et un délai critique  $D_i$ .

La période correspond au délai constant entre deux activations de la tâche. Durant chaque activation, la durée d'exécution de l'instance de la tâche  $i$  est bornée par sa capacité. Le délai critique correspond à la durée maximum admise entre l'activation de la tâche  $i$  et la fin de son exécution. La fin d'exécution d'une tâche  $i$  est déterminée en calculant son temps de réponse  $r_i$ .  $r_i$  correspond au délai entre la date de première activation de  $i$  et l'instant de sa terminaison. Les contraintes temporelles d'un système temps réel sont respectées si, par exemple, pour toutes les tâches  $i$ ,  $r_i \leq D_i$ .

La théorie des files d'attente permet d'analyser les performances de systèmes composés de serveurs, de clients et de zones d'attente : des personnes dans la salle d'attente d'un médecin, un routeur réseau aiguillant des données... Si de nouveaux clients arrivent dans le système alors que le serveur est occupé, leur requête est placée dans une file. La connaissance du taux d'arrivée moyen et du nombre moyen de requêtes que le serveur peut traiter, permet de prédire, entre autres, le nombre moyen, le temps d'attente moyen et la probabilité d'avoir un nombre donné de clients dans la file.

Une file d'attente est décrite par au moins 3 caractéristiques (notation de KENDALL) :  $A|B|c$ .  $A$  représente la fréquence d'arrivée des clients.  $B$  décrit la distribution du temps de service. Finalement,  $c$  est le nombre de serveurs. Le taux d'arrivée et le temps de service ont des distributions qui peuvent être de type Déterministe  $D$ , Markovienne  $M$  ou Générale  $G$ .

## 3 La loi de service $P$

Dans cette section, nous proposons une loi de service  $P$  qui permet d'appliquer les résultats de la théorie des files d'attente aux tampons des sous-systèmes mono-processeur étudiés.

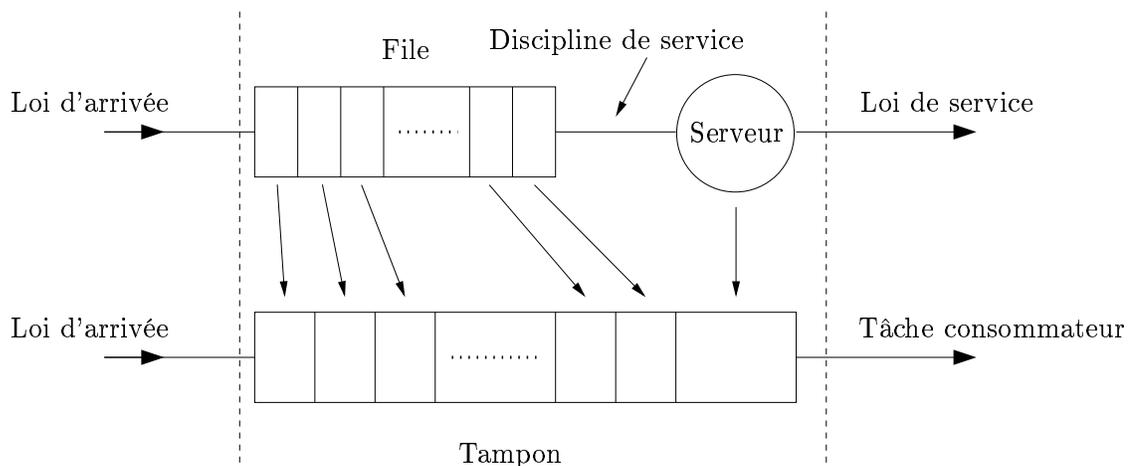


FIG. 1 – Modélisation du tampon par une file d'attente

Les tampons des systèmes que nous étudions sont modélisés par des files d'attente où les clients sont les messages servant aux communications entre tâches. La principale difficulté vient du fait que les tâches consommateurs de notre sous-système ne sont pas activés sur réception de message dans le tampon. En effet, le serveur d'une file d'attente traite un message dès qu'il arrive dans la file. Le serveur de la file ne peut donc pas directement modéliser la tâche consommateur.

Les éléments constituant une file d'attente, c'est à dire, la file et le serveur, modélisent un tampon (cf. figure 1). Les dates de sortie des messages de la file d'attente sont égales aux dates de consommation de la tâche consommateur. Le temps passé dans le tampon est donc égal au temps passé dans la file et le serveur. Nous pouvons faire la même observation pour le taux d'occupation.

Nous proposons donc une loi de service  $P$  qui décrit le comportement temps réel et périodique de la tâche consommateur ordonnancée selon un algorithme préemptif à priorité fixe. Cette loi est décrite par le temps moyen de service et la variance sur ce délai. **Ces deux paramètres sont nécessaires pour obtenir les critères de performances de la théorie des files d'attente.**

Afin de simplifier l'expression des critères de performances et des preuves fournies par la suite, nous considérons qu'un seul message, au maximum, est produit ou consommé par une tâche chaque période. En outre, pour chaque instance  $i$  d'une tâche  $j$ , cette consommation intervient à la fin du temps de réponse, soit  $r_j^i$ .

La loi de service  $P$  est caractérisée par son temps de service moyen  $W_s$ . Nous rappelons la définition générale du temps de service d'une file d'attente :

**Définition 1 (Temps de service)** *Le temps de service  $S_i$  est le temps passé par un client à être traité par un serveur [Kle75].  $S_i$  est donc le temps entre l'activation du serveur et la fin du traitement du message.*

- *Lorsque le système est vide, le serveur est activé dès qu'un nouveau client arrive.*
- *Lorsque un ou plusieurs clients sont en attente dans la file, le serveur est activé immédiatement après la fin du traitement du client précédent.*

Nous transposons cette définition aux tampons étudiés. Les dates de sortie des messages de la file d'attente sont obtenues grâce aux dates de consommation de la tâche consommateur. Finalement, le temps de service de la loi  $P$  est donné par le Théorème (1).

**Théorème 1 (La loi  $P$  : temps de service)** *Lorsque le serveur est occupé, le  $i^{\text{ème}}$  temps de service de la loi  $P$  est égal à :*

$$S'_i = r_{cons}^j + P_{cons} - r_{cons}^{j-1}$$

*Dans le cas contraire,  $S_i$  est égal à :*

$$S''_i = r_{cons}^j + O_j^i$$

*Où  $P_{cons}$  est la période de la tâche consommateur,  $r_{cons}^j$  est le temps de réponse de l'instance  $j$  de la tâche consommateur et  $O_j^i$  est le délai entre la date d'arrivée du message  $i$  et la  $j^{\text{ème}}$  activation de la tâche consommateur.  $O_j^i$  appartient à l'intervalle  $[-r_{cons}^j; P_{cons} - r_{cons}^{j-1}]$ .*

La preuve de ce théorème peut être trouvée dans [Leg04].

## 4 Critères moyens de performance de la file M/P/1

Dans cette section, nous étudions les tampons partagés par  $n$  flux d'arrivées aléatoires de messages et une tâche périodique qui consomme ces messages. Selon la notation de KENDALL, un tel tampon est modélisé par une file d'attente M/P/1 [Kle75, Rob90]. La discipline de service des messages est la politique FIFO<sup>1</sup>.

Nous proposons une approximation de la file M/P/1. Il est nécessaire d'évaluer le temps de service moyen  $W_s$  et la variance  $\sigma_s^2$  du temps de service. Grâce à ces deux paramètres, nous pouvons utiliser les critères de performance de la théorie des files d'attente [Rob90, Kle75].

### 4.1 Notions préliminaires

Préalablement à l'étude du temps de service et de sa variance pour la file M/P/1, nous présentons quelques définitions.

La tâche consommateur est une tâche périodique indépendante ordonnancée suivant un algorithme d'ordonnancement temps réel. Le consommateur est donc activé indépendamment des arrivées de messages dans le tampon. Dans ce cas, une consommation peut être effective ou non.

**Définition 2 (Consommation effective)** *Une consommation est dite :*

- *Effective si la tâche consommateur est activée et au moins un message est présent dans le tampon.*
- *Non-effective lorsque la tâche consommateur est activée et qu'il n'y a pas de message dans le tampon.*

De la Définition (2), nous pouvons déduire un résultat important pour déterminer le temps de service moyen : le taux de consommations effectives.

**Théorème 2** *Le taux de consommations effectives  $U_c$  de la file M/P/1 est égal à :*

$$U_c = 1 - P_0 = \rho$$

Où  $\rho$  est le taux d'utilisation de la file et  $P_0$ , la probabilité que le tampon soit vide.

#### Eléments de preuve :

En effet, une consommation est effective si le tampon est non vide. Si  $P_0$  représente la probabilité de n'avoir aucun message dans le tampon, le taux de consommation effective  $U_c$  est alors de  $1 - P_0$ . Or, pour une file G/G/1,  $\rho = 1 - P_0$  [Kle75]. La file M/P/1 étant un cas particulier de la file G/G/1,  $U_c$  est égale à  $\rho$ .  $\square$

---

<sup>1</sup>Le premier message arrivé est aussi le premier servi.

## 4.2 Proposition de résolution approchée de la file M/P/1

La résolution de la file M/P/1 consiste à trouver le temps de service moyen et la variance sur ce temps de service. Elle nécessite une analyse sur un intervalle de temps infini.

Tout d'abord, nous résolvons le cas où le serveur est requis à chaque activation de la tâche consommateur. Dans ce cas,  $\rho$  (ou  $U_c$ ) tend vers 1 et nous considérons que toutes les consommations sont effectives.

**Théorème 3** *Lorsque  $\rho$  tend vers 1, le temps de service est de :*

$$S'_i = r_{cons}^i + P_{cons} - r_{cons}^{i-1}$$

Où  $P_{cons}$  est la période de la tâche consommateur et  $r_{cons}^i$  est le temps de réponse de l'instance  $i$  de la tâche consommateur.

Le temps de service moyen est égal à :

$$W_s = P_{cons}$$

### Eléments de preuve :

En effet, lorsque  $\rho$  tend vers 1, toutes les consommations sont effectives (cf. Théorème (2)). Comme nous l'avons indiqué ci-dessus, l'analyse de la file M/P/1 doit être faite sur une durée infinie. L'ensemble des  $S_i$  à considérer dans le calcul de  $W_s$  est donc infini. Afin de simplifier le calcul du temps de service moyen, nous réduisons la période d'étude infinie de la file à une durée finie.

Grâce à la périodicité des tâches, les séquences de temps de réponse de leurs instances sont cycliques. La taille commune de toutes ces séquences est la période d'étude<sup>2</sup> du jeu de tâches.

Nous considérons donc un intervalle de temps de  $nc$  fois la période d'étude du jeu de tâches. Il y a  $n$  instances de la tâche consommateur dans cette période d'étude. Chacun des  $n$  temps de service s'exprime alors sous la forme suivante :

- $nc * S'_1 = nc * (r_{cons}^1 + P_{cons} - r_{cons}^n)$
- $nc * S'_2 = nc * (r_{cons}^2 + P_{cons} - r_{cons}^1)$
- $nc * S'_3 = nc * (r_{cons}^3 + P_{cons} - r_{cons}^2)$
- ...
- $nc * S'_{n-1} = nc * (r_{cons}^{n-1} + P_{cons} - r_{cons}^{n-2})$
- $nc * S'_n = nc * (r_{cons}^n + P_{cons} - r_{cons}^{n-1})$

---

<sup>2</sup>La période d'étude (ou hyperpériode) représente une séquence infiniment répétée de l'ordonnancement [LM80].

Finalement, la moyenne de l'ensemble de ces temps de service est égal à :

$$W_s = \sum_1^n S_i = \frac{1}{nc.n} nc.n.P_{cons} = P_{cons}$$

□

Nous étudions maintenant la file lorsque  $\rho$  tend vers 0 ( $U_c$  tend alors vers 0). Dans ce cas, seules quelques rares activations de la tâche consommateur impliquent une requête au serveur. Le nombre de consommations effectives tend également vers 0.

**Théorème 4** *Lorsque  $\rho$  tend vers 0, le temps de service est de :*

$$S_i'' = \frac{S_i'}{2}$$

*Le temps de service moyen est égal à :*

$$W_s = \frac{P_{cons}}{2}$$

### Eléments de preuve :

Lorsque  $\rho$  tend vers 0, les consommations effectives sont rares (cf. Théorème 2). La principale difficulté est de savoir si une activation donnée de la tâche consommateur implique une requête au serveur de la file M/P/1 et donc un nouvel  $S_i$ . Obtenir une valeur exacte de  $W_s$  est complexe, nous proposons donc une approximation de celle-ci.

Désormais, la probabilité que le serveur soit activé sur réception de message est très importante. Le temps de service est donc de la forme  $S_i'' = r_{cons}^j + O_j^i$  (cf. Théorème (1)). Un premier problème consiste à trouver la valeur de  $O_j^i$ . En outre, étant donné que le nombre de requêtes au serveur (ou  $S_i$ ) est inférieur au nombre d'activation du consommateur, l'ensemble des  $S_i$  à prendre en compte dans le calcul de  $W_s$  est difficile à déterminer.

Pour résoudre ces problèmes, nous étudions la probabilité d'avoir un délai particulier entre 2 arrivées successives suivant un processus de poisson. Cette probabilité est représentée par la fonction suivante [Rob90, Kle75] :

$$f(\lambda, t) = 1 - e^{-\lambda t} \tag{1}$$

Lorsque  $\rho$  tend vers 0, la période de la tâche consommateur étant fixée,  $\lambda$  tend vers 0. Dans ce cas, l'intervalle de temps moyen entre 2 arrivées successives tend vers l'infini et la fonction  $f(0, t)$  tend vers 0 quelle que soit la valeur de  $t$ .

La probabilité d'avoir un délai particulier entre 2 arrivées consécutives est la même quel que soit ce délai. Par conséquent, la date d'arrivée intervenant dans l'intervalle<sup>3</sup>  $[0, S_i']$  avec une probabilité uniformément répartie implique que :

$$S_i'' = \frac{S_i' + 0}{2} = \frac{S_i'}{2}$$

---

<sup>3</sup>Le serveur sera nécessairement sollicité durant l'intervalle  $[0, S_i']$ .

En outre, si les dates d'arrivées sont uniformément réparties sur une durée infinie, le nombre d'occurrences de temps de service  $nc$  est le même pour tous les  $S_i$ . Le temps de service moyen est donc de :

$$W_s = \frac{1}{n * nc} \sum_{i=1}^n \frac{nc * S'_i}{2}$$

Soit,

$$W_s = \frac{1}{2n} \sum_{i=1}^n S'_i$$

Finalement, nous avons :

$$W_s = \frac{P_{cons}}{2}$$

□

Les deux cas précédents sont des extrêmes ( $\rho$  est compris entre 0 et 1). Grâce à une régression linéaire, nous obtenons l'approximation du temps de service moyen de la file M/P/1 et de sa variance valide quelle que soit la valeur de  $\rho$ .

**Théorème 5** *Le temps de service de la file M/P/1 est de :*

$$S_i = \rho S'_i + (1 - \rho).S''_i = (1 + \rho).S'_i$$

*Le temps de service moyen est alors égal à :*

$$W_s = \frac{P_{cons}}{2}(1 + \rho) = \frac{P_{cons}}{2(1 - \lambda \frac{P_{cons}}{2})}$$

*Et la variance de ce temps de service moyen est de :*

$$\sigma_s^2 = \rho. \left( \frac{1}{n} \sum_{i=1}^n S'_i - W_s^2 \right) + (1 - \rho). \frac{P_{cons}^2}{12}$$

La démonstration de ce théorème consiste à trouver les poids respectifs du temps de service moyen lorsque  $\rho$  tend vers 0 et 1 et à appliquer ensuite une regression linéaire du premier ordre (une preuve détaillée de ce théorème peut être trouvée dans [Leg04]).

## 5 Critères maximums de performance de la file P/P/1

Les critères moyens de performance permettent d'évaluer les caractéristiques du tampon de manière à diminuer le risque de débordement. Dans cette section, nous exhibons des critères maximums de performance qui garantissent de façon déterministe le bon fonctionnement de l'application.

Ces critères concernent les tampons partagés par  $n$  tâches producteurs et une tâche consommateur. Selon la notation de KENDALL, un tel tampon est modélisé par un file d'attente P/P/1. La discipline de service des messages est toujours la politique FIFO.

Il existe des similitudes entre notre système producteur/consommateur et certains systèmes issus du monde des réseaux hauts débits. C'est le cas des services de communication utilisés par le transport de la voix dans les réseaux ATM : la couche d'adaptation AAL1 [GK96].

Nous nous sommes donc basés sur les résultats obtenus dans le cadre des réseaux ATM afin de proposer des bornes maximums pour le taux d'occupation et le temps d'attente des messages lorsque les arrivées de messages sont déterministes.

Dans le service de communication AAL1 d'ATM, un producteur envoie des données audios à une cadence fixe  $d$ . Ce débit est exprimé en nombre de cellules émises par seconde, l'unité utilisée dans les réseaux ATM. Un délai variable mais borné est nécessaire à chaque cellule pour atteindre le consommateur. La cadence de consommation des cellules doit être identique à celle d'émission. Chaque cellule est donc stockée dans un tampon pendant un laps de temps suffisant pour supprimer la gigue induite par le réseau de transmission. Dans [GK96], il est démontré que le taux d'occupation du tampon est dans ce cas borné conformément au Théorème (6) ci-après :

**Théorème 6 (Taille maximum des tampons ATM)** *La taille maximum des tampons régulant le trafic de la couche de transport à débit fixe d'ATM est de :*

$$L_{max} = \left\lceil \frac{W_{max}}{d} \right\rceil \quad (2)$$

où  $W_{max} = 2.\Delta$ .

$W_{max}$  constitue dans ce cas le **plus grand délai pendant lequel les informations peuvent s'accumuler dans le tampon**.

Les systèmes que nous étudions possèdent des similitudes avec le système décrit précédemment. Nous adaptons donc le Théorème (6) afin de proposer des bornes sur le temps d'attente maximum et le taux d'occupation des tampons partagés par des tâches périodiques dont la priorité est fixe. Nous supposons que le taux d'arrivée est déterminé à partir du plus petit délai entre deux arrivées de messages. Dans le cas d'un tampon partagé par  $n$  producteurs périodiques et un consommateur périodique, la borne maximum sur le taux d'occupation du tampon est donné par [LSN<sup>+</sup>03] :

**Proposition 1** *La borne sur le taux d'occupation  $L_{max}$  d'un tampon P/P/1 est de :*

$$L_{max} = \max_{\forall y \geq 0} \left( \sum_{prod \in PROD} \left\lceil \frac{W_{max} + O_{prod}}{P_{prod}} \right\rceil - y \right) \quad (3)$$

Où  $PROD$  est l'ensemble des tâches producteurs,  $P_{prod}$  est la période du producteur  $prod$  et  $O_{prod}$  est le délai maximum entre les dates d'activation du consommateur et du producteur  $prod$ .

La borne maximum sur la taille du tampon est obtenue par construction à partir du délai maximum d'accumulation des messages et du débit maximum. Selon la configuration du jeu de tâches et compte tenu du fait que le tampon fonctionne de façon FIFO, entre l'instant de production d'un message par un producteur  $i$  et l'instant de sa consommation,  $y$  activations du (ou des) consommateur(s) sont nécessaires pour consommer les messages déjà présents dans le tampon. Nous ne cherchons pas, pour l'instant, à quantifier précisément  $y$ .

Le délai d'attente maximum  $W_{max}$  est donné par la Définition (3) ci-dessous :

**Définition 3 (Délai d'attente maximum)** *Le délai d'attente maximum d'un message est de*

$$W_{max} = (y + 1).P_{cons} + D_{cons} \quad (4)$$

Où  $y$  est le nombre de messages déjà présents dans le tampon au moment où le message  $i$  arrive.

Finalement, grâce à une étude aux limites par rapport à  $y$  de l'Equation (3).

**Théorème 7** *Pour un tampon P/P/1 partagé par un jeu de tâches harmoniques<sup>4</sup> avec  $n$  tâches producteurs où  $D_i \leq P_i$ , les bornes maximums sur le taux d'occupation et sur le temps d'attente sont de :*

$$L_{max} = 2.n \tag{5}$$

$$W_{max} = 2.n.P_{cons}$$

*Dans le cas d'un jeu de tâches non harmoniques, les bornes maximums sur le taux d'occupation et sur le temps d'attente sont de :*

$$L_{max} = 2.n + 1 \tag{6}$$

$$W_{max} = (2.n + 1).P_{cons}$$

La preuve du Théorème (7) est disponible dans [Leg04].

## 6 Simulation des files M/P/1 et P/P/1

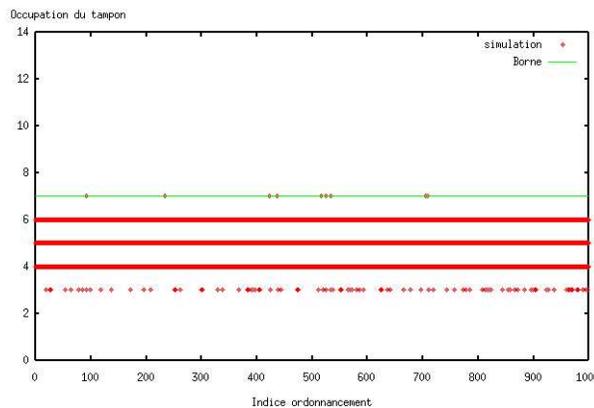


FIG. 2 – Analyse pire cas : borne maximum sur le taux d'occupation de P/P/1

Dans cette section, nous présentons les résultats des simulations menées afin d'évaluer les critères de performance proposés.

En ce qui concerne l'analyse "pire cas" (les files d'attente P/P/1), 1000 jeux de trois tâches producteurs périodiques et d'une tâche périodique consommateur ont été générés. Pour chaque tampon, 1000 ordonnancements sont simulés sur une durée de 20 fois la période d'étude du jeu de tâches. Sur la figure 2, la borne sur le taux d'occupation maximum proposée par le Théorème (7) est comparée aux résultats de simulation. Les résultats montrent que la borne est parfois atteinte mais n'est jamais dépassée.

Pour l'analyse "moyen cas" (les files d'attente M/P/1), 900 systèmes composés de  $n$  flux d'arrivée de messages et d'une tâche consommateur périodique ont été générés. Les durées de simulation sont de 20 000 000 000 unités de temps. Nous présentons les résultats concernant les tâches périodiques de forte et moyenne/basse priorité. Le taux d'utilisation de la file varie de 0.01 à 0.90. La période des tâches est inférieure à 1000 unités de temps.

<sup>4</sup>Un ensemble de tâches est harmonique si toutes les tâches possèdent des périodes multiples entre elles.

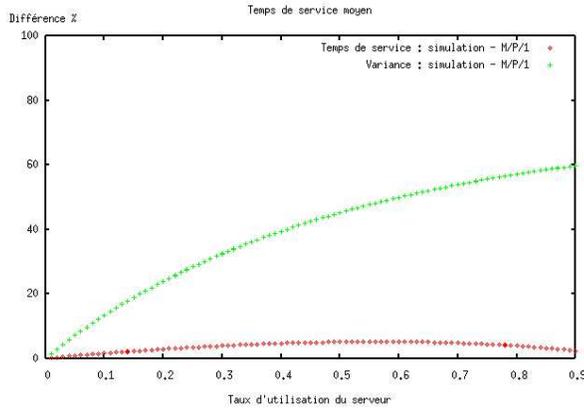


FIG. 3 – Temps de service moyen et variance pour la file M/P/1 : forte priorité

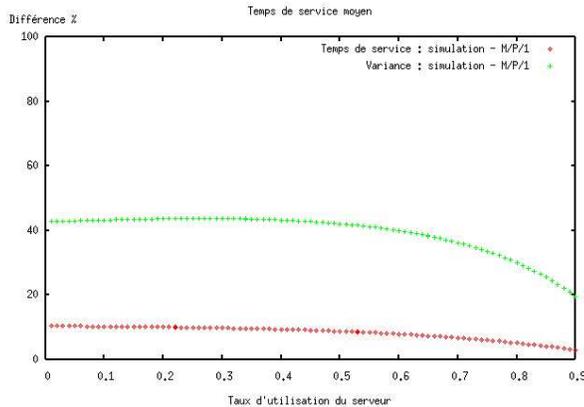


FIG. 4 – Temps de service moyen et variance pour la file M/P/1 : priorité basse/moyenne

Les courbes des figures 3 et 4 représentent la différence en pourcentage entre le temps de service moyen (resp. la variance sur le temps de service moyen) proposé par le Théorème (5) et les valeurs mesurées au cours de la simulation de la file M/P/1.

Sur la première figure, la tâche consommateur possède une forte priorité. Nous observons que le temps de service théorique est relativement proche des résultats de simulation. La différence varie entre 0 et 5%. Par contre, la variance théorique donne de bons résultats uniquement lorsque le taux d'utilisation du serveur est faible. Plus ce taux est important, moins notre variance théorique est proche de la variance de la file M/P/1 simulée. La différence varie entre 0 et 60%.

Sur la deuxième figure, la tâche consommateur possède une priorité faible. Dans ce cas, le temps de service théorique est légèrement moins bon que précédemment. La différence varie entre 0 et 10%. La variance théorique est proche des résultats de simulation lorsque le taux d'utilisation du serveur de la file M/P/1 tend vers 1. Lorsque ce taux tend vers 0, les résultats se dégradent : la différence varie entre 0 et 43%.

D'autre part, nous proposons de comparer les valeurs du taux d'occupation et du temps d'attente mesurées sur la simulation de la file M/P/1 avec les critères théoriques des files M/G/1, M/M/1 et M/D/1. Ces critères nécessitent les valeurs du temps de service moyen et éventuellement de la variance de ce temps de service moyen.

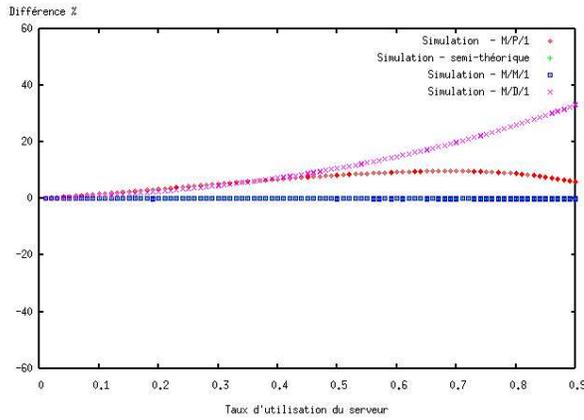


FIG. 5 – M/P/1 : taux d'occupation avec un consommateur de forte priorité

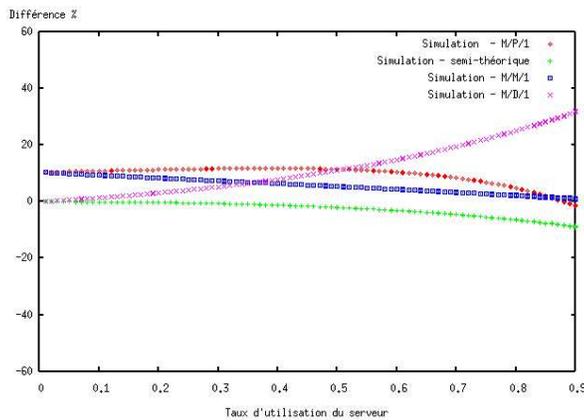


FIG. 6 – M/P/1 : taux d'occupation avec un consommateur de priorité basse/moyenne

Afin de calibrer notre simulateur, nous calculons la différence entre la valeur des critères mesurés au cours de la simulation et la valeur des critères que nous désignons par le terme "semi-théorique". La valeur de ces critères est obtenue en appliquant aux critères de la file d'attente M/G/1 le temps de service moyen et la variance sur ce temps de service issus de la simulation. La qualité de la simulation est inversement proportionnelle à cette différence.

Nous indiquons systématiquement la différence entre la valeur de critères mesurés au cours de la simulation et la valeur des critères de la file M/G/1 auxquels nous appliquons les équations de  $W_s$  et  $\sigma_s^2$  proposées par le Théorème (5). La courbe qui en résulte permet d'évaluer nos propositions (cf. figures 5 et 6).

Nous calculons également la différence entre la valeur de critères mesurés au cours de la simulation et la valeur des critères des files M/M/1 et M/D/1 auxquels nous appliquons le temps de service moyen proposé (ces files ne nécessitent que ce paramètre).

Etant donné que le taux d'occupation et le temps d'attente moyen sont liés par la loi de LITTLE [Kle75, Rob90], les observations que nous faisons sont valables, dans les conditions données, quel que soit le critère considéré. Les courbes concernant le temps d'attente moyen sont d'ailleurs similaires à celle du taux d'occupation moyen.

En dépit des résultats de la variance, les critères théoriques de la file M/P/1 sont proches des résultats de simulation. La différence varie entre 0 et 10% quel que soit la priorité du consommateur. Ces critères

théoriques sont les critères de la file M/G/1 ou M/M/1 auxquels sont appliqués les  $W_s$  et  $\sigma_s^2$  proposés par le Théorème (5).

## 7 Conclusions et perspectives

Cet article présente une analyse de performance d'un système temps réel mono-processeur composé de tâches périodiques indépendantes. Les tâches sont ordonnancées selon un algorithme à priorités fixes [LL73] et accèdent à des tampons.

Nous nous concentrons, d'une part, sur le respect des contraintes temporelles des tâches, et d'autre part, sur les critères de performance, tels que le temps d'attente ou le taux d'occupation des tampons. L'ordonnancement temps réel propose beaucoup de tests de faisabilité pour vérifier les échéances de tâches indépendantes [JP86, ABRT93]. Par contre, peu de résultats existent pour l'analyse de performance des tampons [Kre00].

Nous proposons donc une loi de service, la loi  $P$ . La loi  $P$  permet de modéliser le comportement des tâches périodiques ordonnancées par un algorithme à priorités fixes. Elle considère également que l'activation des tâches est indépendante des arrivées de messages. Par conséquent, les tests de faisabilité concernant les tâches indépendantes peuvent être utilisés.

A partir de cette loi  $P$ , deux files d'attente sont étudiées : la file M/P/1 et la file P/P/1. Les systèmes contenant des tampons peuvent être étudiés selon une analyse pire cas ou moyen cas. L'analyse pire cas concerne les tampons dont le délai minimum entre deux arrivées de messages est connu, on utilise alors les critères de performance de la file P/P/1. Si aucune hypothèse sur ce délai inter-arrivée n'est donnée, une analyse moyen cas peut tout de même être réalisée ; les critères de performance de la file M/P/1 sont alors utilisés.

Une résolution exacte de la file P/P/1 est donnée et nous proposons une résolution approchée pour la file M/P/1. L'approximation de la file M/P/1 est basée sur le fait que, d'une part, le système tend vers un comportement déterministe lorsque le taux d'utilisation de la file est important, et d'autre part vers un comportement uniforme lorsque le taux est faible.

Les files P/P/1 et M/P/1 sont également évaluée grâce à des simulations. Les résultats de simulations montrent qu'en utilisant le temps de service moyen et la variance sur ce temps de service, la différence entre les temps d'attente et taux d'occupation moyens théoriques et simulés est comprise entre 0 et 10%.

Ces résultats restent à généraliser, notamment lorsque plusieurs tâches consomment des messages dans un même tampon ou lorsque les délais critiques sont arbitraires. En outre, il faut étudier d'autres critères de performance afin d'avoir une analyse plus fine des tampons. Enfin, nous n'avons pas abordé le cas où les messages possèdent des contraintes temporelles. La faisabilité consisterait, non seulement, à étudier les tâches et les tampons de l'application, mais également, à vérifier les contraintes des messages.

## 8 Remerciements

Nous voudrions remercier Gégardo Rubino (IRISA/INRIA Rennes) pour son aide dans le travail présenté dans cet article.

## Références

- [ABRT93] A. N. Audsley, A. Burns, M. Richardson, and K. Tindell. Applying new scheduling theory to static priority pre-emptive scheduling. *Software Engineering Journal*, pages 284–292, 1993.
- [Ari97] Arinc. *Avionics Application Software Standard Interface*. The Arinc Committee, January 1997.
- [GK96] M. Gagnaire and D. Kofman. *Réseaux Haut Débit : réseaux ATM, réseaux locaux, réseaux tout-optiques*. Masson-Inter Editions, Collection IIA, 1996.
- [JP86] M. Joseph and P. Pandya. Finding Response Time in a Real-Time System. *Computer Journal*, 29(5) :390–395, 1986.
- [Kle75] L. Kleinrock. *Queueing Systems : theory*. Wiley-interscience, 1975.
- [Kre00] J. Kreimer. Real Time System with homogeneous servers and noidentical channels in steady-state. *Computers and Operations Research*, (29) :1465–1473, December 2000.
- [Leg04] J. Legrand. *Contribution à l'ordonnancement des systèmes temps réel comprenant des tampons*. Thèse de doctorat, Département informatique, Université de Bretagne Occidentale, Brest, 2004.
- [LL73] C. L. Liu and J. W. Layland. Scheduling Algorithms for Multiprogramming in a Hard Real-Time Environment. *Journal of the Association for Computing Machinery*, 20(1) :46–61, January 1973.
- [LM80] J.Y.T Leung and M.L. Merril. A note on preemptive scheduling of periodic real time tasks. *Information processing Letters*, 3(11) :115–118, 1980.
- [LSN<sup>+</sup>03] J. Legrand, F. Singhoff, L. Nana, L. Marcé, F. Dupont, and H. Hafidi. About Bounds of Buffers Shared by Periodic Tasks : the IRMA project. In the 15th Euromicro International Conference of Real Time Systems (WIP Session), Porto, July 2003.
- [RCR01] P. Richard, F. Cottet, and M. Richard. On line Scheduling of Real Time Distributed Computers With Complex Communication Constraints. 7th Int. Conf. on Engineering of Complex Computer Systems, Skovde (Sweden), June 2001.
- [Riv98] N. Rivierre. *Ordonnancement temps réel centralisé, les cas préemptifs et non préemptifs*. Thèse de doctorat, Université de Versailles Saint Quentin, février 1998.
- [Rob90] T. G. Robertazzi. *Computer Networks and Systems : queueing theory and performance evaluation*. Springer-Verlag, 1990.