

Formation Automatique et Informatique Industrielle

Master 1 S2

Matière : Systèmes Embarqués et Systèmes
Temps Réel SE-STR

Par : ATOUI Hamza

Plan du cours

- Politiques d'ordonnancement:
 - Partie 3 : ordonnancement des tâches apériodiques à contrainte souple dans un contexte de tâches périodiques à contrainte stricte.
 - **Traitement en arrière plan.**
 - **Traitement par serveur:**
 - Serveur par scrutation (polling).
 - Serveur sporadique

Introduction

- Pour tenir compte de tâches apériodiques parmi l'ensemble des tâches à ordonnancer, il existe plusieurs approches.
- L'objectif est toujours d'ordonnancer les tâches apériodiques à contrainte souple dans un contexte de tâches périodiques à contrainte stricte.
- Parmi les approches existantes, deux sont couramment employées :
 - Le traitement en arrière plan.
 - Le traitement par serveur.

Politiques d'ordonnancement

TRAITEMENT EN ARRIÈRE PLAN

Traitement en arrière plan

- La priorité est donnée aux tâches périodiques par une des politiques qu'on a étudié {'RM', 'DM', 'EDF', 'LLF'}. Quand le processeur est **oisif (IDLE STATE)** les tâches apériodiques prennent la main pour exécuter par une des politiques suivantes {'FCFS', 'SJF', 'SRT', 'PB'}.
- traitement le plus simple est d'utiliser le FCFS , mais reste le moins performant.

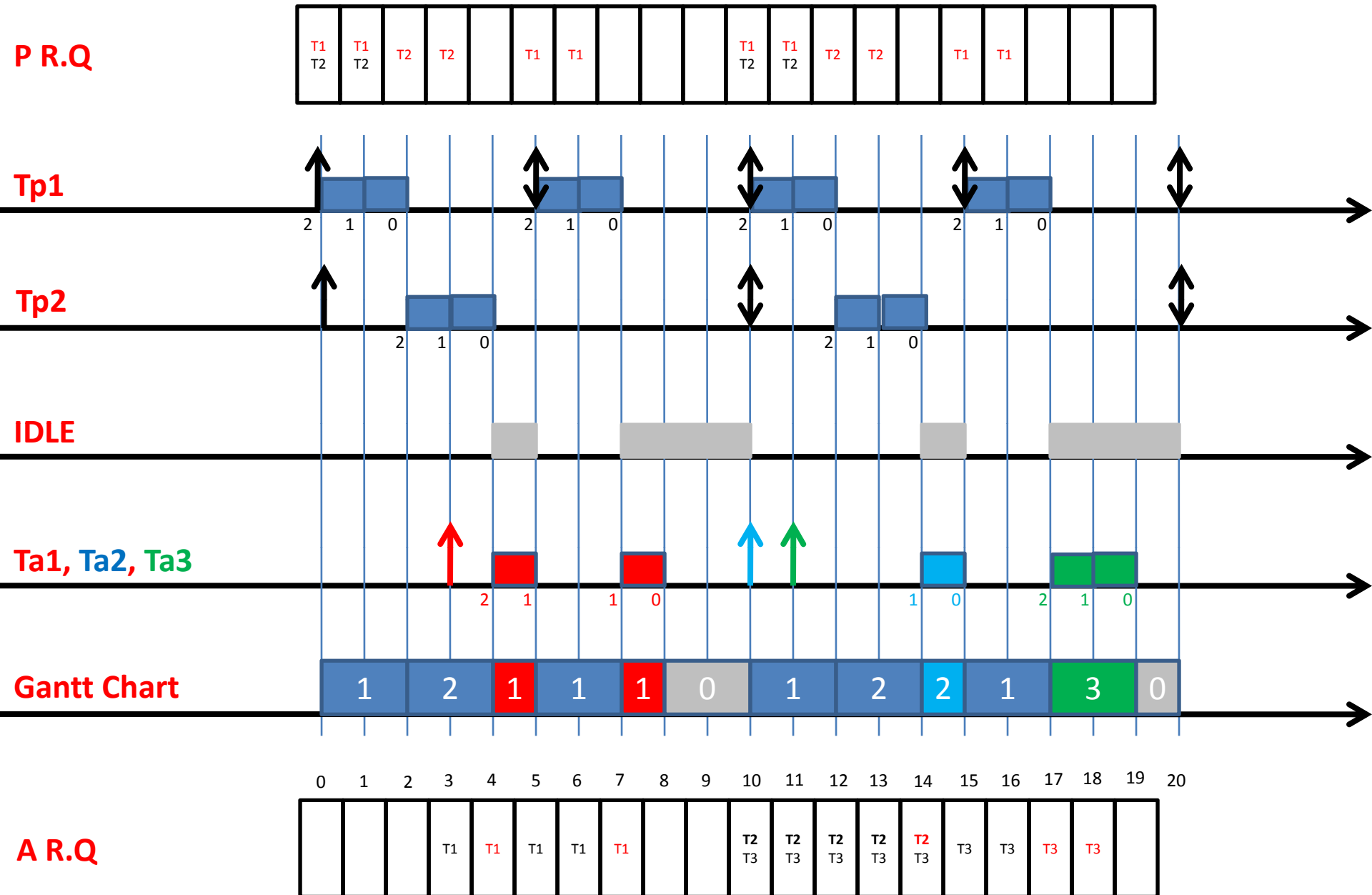
Traitement en arrière plan

- Exemple : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=2, P=5)$.
 - $Tp2(r0=0, C=2, P=10)$.
- Tâches apériodiques :
 - $Ta1(r0=3, C=2)$.
 - $Ta2(r0=10, C=1)$.
 - $Ta3(r0=11, C=2)$.
- Ordonnancer ce système par RM+FCFS (T.A.P).
- Calculer le AWT et ATT pour les tâches apériodiques.

Traitement en arrière plan

- Suivi toutes les étapes pour ordonnancer Tp1 et Tp2 par RM.
- ***Faites attention*** à l'intervalle de simulation, dans ce cas est un multiple de HP (**il faut terminer les capacités des tâches apériodiques**).
- L'ordonnanceur de ce système manipule deux **Ready Queue (R.Q)** une pour les tâches périodiques et l'autre pour les tâches apériodiques.

Traitement en arrière plan



Traitement en arrière plan

Task	r0	C	Completion Date (CD)	Turn Around Time (TAT) (CD – r0)	Waiting Time (WT) (TAT - C)
Task1	3	2	8	8-3 = 5	5-2 = 3
Task2	10	1	15	15-10 = 5	5-1 = 4
Task3	11	2	19	19-11 = 8	8-2 = 6

- $AWT = \text{mean}(WT_i) = \text{mean}(3, 4, 6) = 4.33$.
- $ATT = \text{mean}(TAT_i) = \text{mean}(5, 5, 8) = 6.00$.

Traitement en arrière plan

- Exercice : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=2, P=20)$.
 - $Tp2(r0=0, C=1, P=5)$.
 - $Tp3(r0=0, C=1, P=10)$.
- Tâches apériodiques :
 - $Ta1(r0=1, C=5)$.
 - $Ta2(r0=2, C=2)$.
 - $Ta3(r0=6, C=4)$.
- Ordonnancer ce système par :
 - RM+SJF (T.A.P).
 - RM+FCFS (T.A.P).
- Calculer le AWT et ATT pour les tâches apériodiques.

Politiques d'ordonnancement

TRAITEMENT PAR SERVEUR

Introduction

- Le principe est de **définir une tâche périodique spéciale**, appelée **serveur apériodique**, active durant son temps d'exécution (appelé capacité du serveur) les tâches apériodiques en attente.
- C'est une forme de **réservation garantie de temps CPU** consacré uniquement aux tâches apériodiques. Deux types de serveurs existent:
 - **Le serveur par scrutation.**
 - **Le serveur sporadique.**

Introduction

- Le serveur est caractérisé par :
 - Sa période.
 - Son temps d'exécution : capacité du serveur.
 - Serveur généralement ordonnancé suivant le même algorithme que les autres tâches périodiques.
 - Une fois actif, le serveur sert les tâches apériodiques dans la limite de sa capacité.
 - l'ordre de traitement des tâches apériodiques ne dépend pas de l'algorithme des tâches périodiques.

Serveur par scrutation (polling)

- A chaque activation, le traitement des tâches en suspens jusqu'a épuisement de la capacité ou jusqu'a ce qu'il n'y ait plus de tâches en attente.
- si aucune tâche n'est en attente (a l'activation ou parce que la dernière tâche a été traitée) , le serveur se suspend immédiatement et perd sa capacité qui peut être réutilisée par les tâches périodiques (amélioration du temps de réponse).

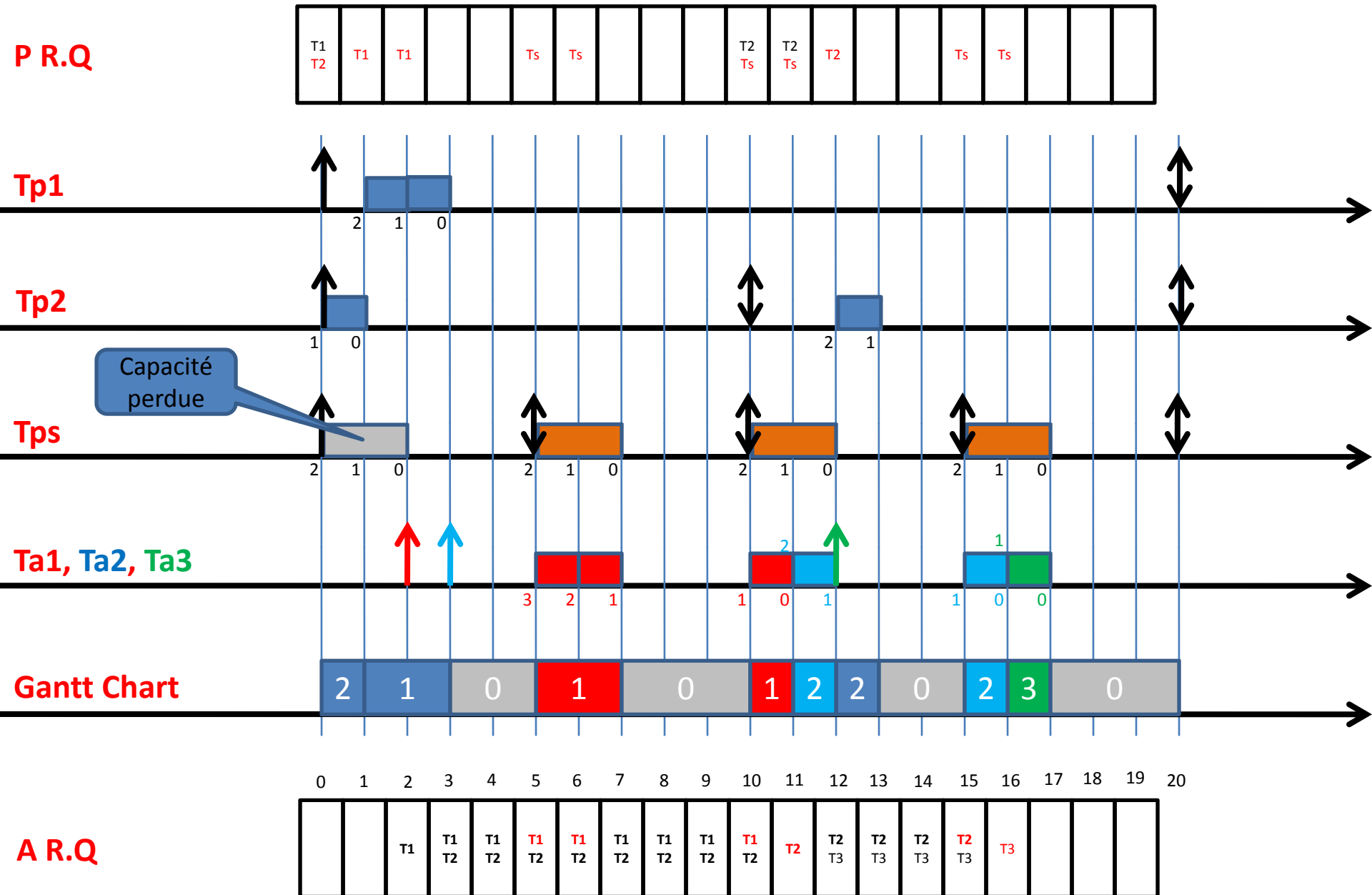
Serveur par scrutation (polling)

- Exemple : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=2, P=20)$.
 - $Tp2(r0=0, C=1, P=10)$.
 - $Tps(r0=0, C=2, P=5)$.
- Tâches apériodiques :
 - $Ta1(r0=2, C=3)$.
 - $Ta2(r0=3, C=2)$.
 - $Ta3(r0=12, C=1)$.
- Ordonnancer ce système par RM+FCFS (S.P.S).
- Calculer le AWT et ATT pour les tâches apériodiques.

Serveur par scrutation (polling)

- Suivi toutes les étapes pour ordonnancer $Tp1$, $Tp2$ et Tps (**Tps seulement si au moins une tâches apériodique arrive avant/avec $r0$ de Tps si Tps est prioritaire**) par RM.
- ***Faites attention*** à l'intervalle de simulation, dans ce cas est un multiple de HP (**il faut terminer les capacités des tâches apériodiques**).
- L'ordonnanceur de ce système manipule deux **Ready Queue (R.Q)** une pour les tâches périodiques et l'autre pour les tâches apériodiques.

Serveur par scrutation (polling)



Serveur par scrutation (polling)

Task	r0	C	Completion Date (CD)	Turn Around Time (TAT) (CD – r0)	Waiting Time (WT) (TAT - C)
Task1	2	3	11	11-2 = 9	9-3 = 6
Task2	3	2	16	16-3 = 13	13-2 = 11
Task3	12	1	17	17-12 = 5	5-1 = 4

- $AWT = \text{mean}(WT_i) = \text{mean}(6, 11, 4) = 7.00$.
- $ATT = \text{mean}(TAT_i) = \text{mean}(9, 13, 5) = 9.00$.

Serveur par scrutation (polling)

- limitations du serveur par scrutation :
 - perte de la capacité si aucune tâche apériodique en attente.
 - si occurrence d'une tâche apériodique alors que le serveur est suspendu, il faut attendre la requête suivante.

Serveur par scrutation (polling)

- Exercice : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=2, P=20)$.
 - $Tp2(r0=0, C=1, P=5)$.
 - $Tps(r0=0, C=3, P=10)$.
- Tâches apériodiques :
 - $Ta1(r0=1, C=4)$.
 - $Ta2(r0=2, C=2)$.
- Ordonnancer ce système par :
 - RM+SRT (S.P.S).
 - RM+FCFS (S.P.S).
- Calculer le AWT et ATT pour les tâches apériodiques.

Serveur sporadique

- Le serveur sporadique améliore le temps de réponse des tâches apériodiques sans diminuer le taux d'utilisation du processeur pour les tâches périodiques.
- le serveur sporadique peut être considéré comme une tâche périodique « normale » du point de vue des critères d'ordonnancement.

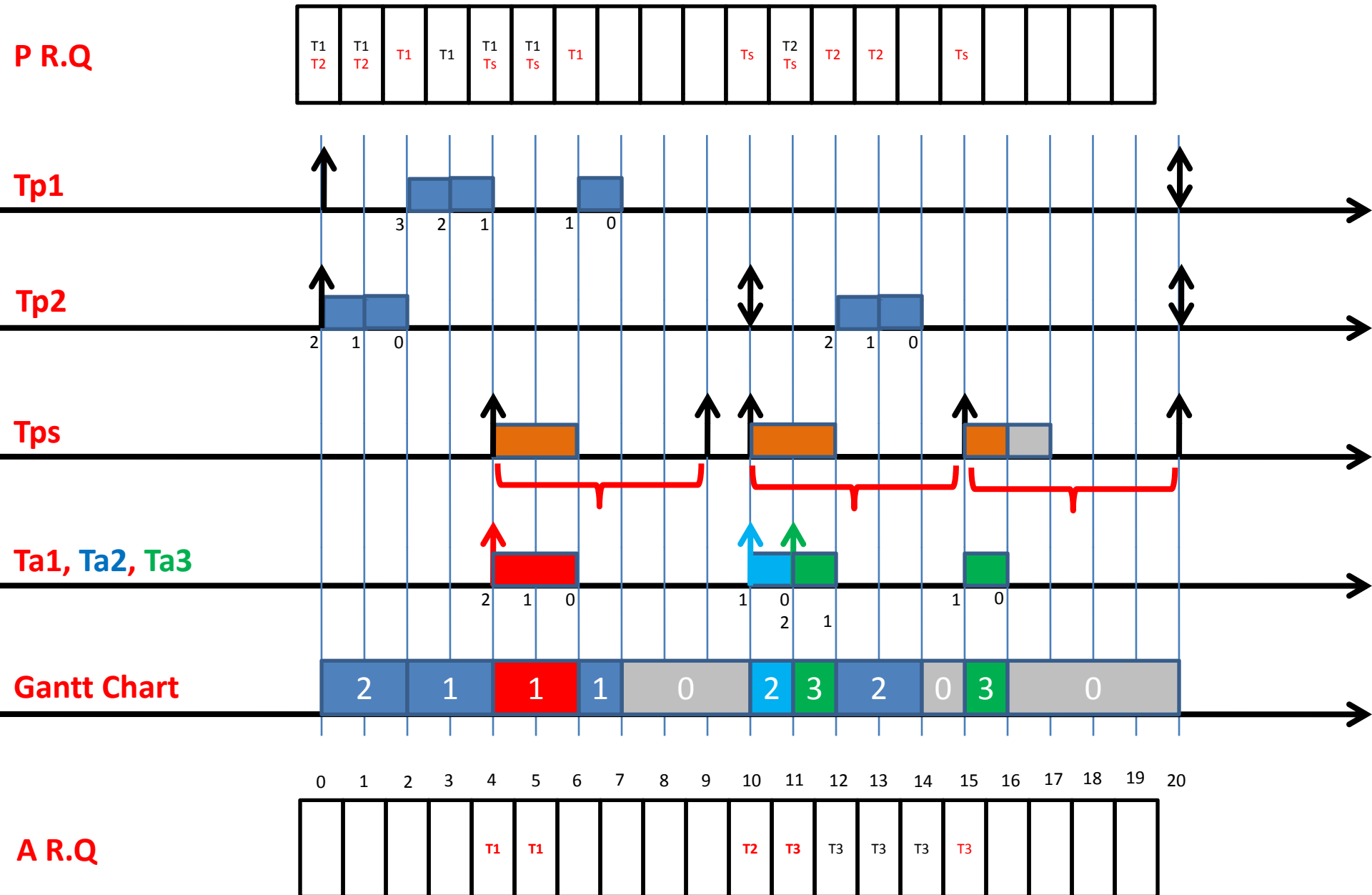
Serveur sporadique

- Si le serveur sporadique est activé par le biais de présence des tâches apériodiques, ce dernier exécute les tâches apériodiques sur le long de son capacité par son période (avec la possibilité de suspendre le serveur si les tâches apériodiques épuisent leurs capacités avant l'écoulement de la capacité du serveur).

Serveur sporadique

- Exemple : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=3, P=20)$.
 - $Tp2(r0=0, C=2, P=10)$.
 - $Tps(C=2, P=5)$.
- Tâches apériodiques :
 - $Ta1(r0=4, C=2)$.
 - $Ta2(r0=10, C=1)$.
 - $Ta3(r0=11, C=2)$.
- Ordonnancer ce système par RM+FCFS (S.S).
- Calculer le AWT et ATT pour les tâches apériodiques.

Serveur sporadique



Serveur sporadique

Task	r0	C	Completion Date (CD)	Turn Around Time (TAT) (CD – r0)	Waiting Time (WT) (TAT - C)
Task1	4	2	6	$6 - 4 = 2$	$2 - 2 = 0$
Task2	10	1	11	$11 - 10 = 1$	$1 - 1 = 0$
Task3	11	2	16	$16 - 11 = 5$	$5 - 2 = 3$

- $AWT = \text{mean}(WT_i) = \text{mean}(0, 0, 3) = 1.00$.
- $ATT = \text{mean}(TAT_i) = \text{mean}(2, 1, 5) = 2.66$.

Serveur sporadique

- Exercice : soit le système temps réel suivant:
- Tâches périodiques :
 - $Tp1(r0=0, C=1, P=5)$.
 - $Tp2(r0=0, C=4, P=15)$.
 - $Tps(C=5, P=10)$.
- Tâches apériodiques :
 - $Ta1(r0=4, C=2)$.
 - $Ta2(r0=8, C=2)$.
- Ordonnancer ce système par :
 - RM+FCFS (S.S).
- Calculer le AWT et ATT pour les tâches apériodiques.