

□Images/logo_SU.jpeg

SORBONNE UNIVERSITÉ

RÉSOLUTION DE PROBLÈME : PROBLÈME
D'ORDONNANCEMENT

Rapport de projet

Encadrant :
Evripidis Bampis

Étudiants :
Yannis Elrharbi-Fleury
Yuan Fangzheng

Table des Matières

1	Introduction	2
2	Structure du programme	3
2.1	Les distributions	3
2.2	Les tâches	3
2.3	Les machines	4
2.4	Les machines parallèles	4
3	Implémentation des algorithmes	5
3.1	Algorithme optimal : Shortest processing time	5
3.2	Exécution par prédiction	5
3.3	Round-Robin	6
3.4	Exécution parallèle	6
3.5	Exécution parallèle dynamique	7
4	Test du bon fonctionnement des modèles	9
4.1	Modèle Prédiction	9
4.2	RR	10
4.3	SPT	11
4.4	Parallèle	11
5	Résultats expérimentaux	13
5.1	Statistiques des tâches	13
5.2	Introduction des dates d'arrivée	13
5.3	Cadre classique	16

1 Introduction

Ce projet porte sur l'étude de solutions au problème d'ordonnancement.

Étant donné N tâches et 1 machine, il s'agit de trouver un ordonnancement de ces tâches minimisant la somme de leur temps de complétude (minimiser le temps d'attente total).

La machine ne connaît pas forcément leur durée d'exécution réelle.

Dans ce rapport, nous étudions et mesurons la qualité de plusieurs solutions en fonction de l'erreur de prédiction.

Le code est écrit en Python et sera fourni en annexe.

2 Structure du programme

Nous avons naturellement opté pour une approche orientée objet du problème.

2.1 Les distributions

La classe *Distribution* représente un ensemble de distributions de probabilité et leurs paramètres.

Lors de son instanciation, elle prend en argument des fonctions permettant de générer un tuple de valeurs.

La méthode *sample* renvoie un tuple contenant :

- une durée réelle
- une durée erreur de prédiction
- un instant d'arrivée

2.2 Les tâches

La classe *Task* représente une tâche, dont les attributs sont générés à partir d'un objet de type *Distribution*.

Elle possède notamment comme attributs :

- un ensemble de durée : durée réelle, durée prédite (générées à partir de *Distribution*)
- un état : *paused*, *running*, *finished*, *not available*
- un curseur *currentStep* permettant d'avancer dans l'exécution de la tâche
- un numéro d'identification

Une tâche possède trois méthodes :

- *hasFinished* : renvoie si la tâche est achevée ou non
- *forward* : exécute la tâche d'un pas de temps, renvoie une exécution de *hasFinished*
- *restart* : réinitialise la tâche à son état initial

2.3 Les machines

Notre idée était de créer une classe *Machine* représentant une machine capable de travailler sur un ensemble de tâches. Les différents algorithmes que nous présenterons dans la partie suivante en héritent.

Chaque machine possède notamment comme attributs :

- des dictionnaires de tâches à différents états : *allTasks*, *workingTasks*, *pausedTasks*, *finishedTasks*
- une vitesse d'exécution
- une horloge donnant le temps de la machine
- une clé d'affichage (une fonction lambda) permettant de trier les tâches de la machine lors de son affichage

Ainsi, *Machine* possède plusieurs méthodes concernant ses tâches : ajouter ou supprimer des tâches; démarrer, mettre en pause ou terminer une tâche.

Elle possède aussi des méthodes permettant de les traiter :

- une méthode de travail *work* faisant travailler les tâches sur un pas de temps de la machine
- une méthode abstraite de traitement *run* traitant les tâches avant chaque étape de travail, elle permet d'introduire l'algorithme de la machine
- une méthode de démarrage *boot* démarrant la machine et la faisant tourner jusqu'à ce que toutes ses tâches soient terminées

2.4 Les machines parallèles

La classe *Parallel* fonctionne de façon analogue à *Machine*, à ceci près qu'elle ne fournit pas le travail aux tâches elle même.

Le travail sur les tâches est effectué par deux machines (*Prediction* et *Round-Robin*) que possède *Parallel*, les exécutant avec une vitesse λ et $1 - \lambda$.

3 Implémentation des algorithmes

Grâce à notre réflexion claire sur les structures de données, nous avons pu facilement implémenter les algorithmes demandés.

Dans cette partie, nous nous contenterons de présenter le fonctionnement de ceux-ci. Nous étudierons leurs performances dans la prochaine section.

Les algorithmes suivant héritent de *Machine* ou *Parallel* et ne font que surcharger la méthode *run*.

3.1 Algorithme optimal : Shortest processing time

L'algorithme *SPT* est un algorithme optimal pour ce problème.

Il connaît la durée réelles des tâches et les exécute de la plus courte à la plus grande.

```
1 def run(self, step):
2     if len(self.workingTasks) == 0:
3         nextTask = sorted(list(self.pausedTasks.values()), key=lambda
4             ↪ x:x.realLength)[0]
5         self.startTask(nextTask)
6     return self.work(step)
```

Sa méthode *run* est claire : si aucune tâche n'est en cours d'exécution, alors la machine exécute la plus courte.

3.2 Exécution par prédiction

Pour l'algorithme *Prediction*, la machine n'a pas accès aux durées réelles des tâches. Elle ne connaît que leur durée prédite, avec par conséquent une certaine erreur.

```
1 def run(self, step):
2     if len(self.workingTasks) == 0:
3         nextTask = sorted(list(self.pausedTasks.values()), key=lambda
4             ↪ x:x.predLength)[0]
5         self.startTask(nextTask)
```

```
5         return self.work(step)
```

La méthode est quasiment identique à l'algorithme précédent, à ceci près que les tâches sont triées en fonction de leur durée prédite.

3.3 Round-Robin

L'algorithme *Round-Robin* partage son travail de façon égale entre les tâches.

Cet algorithme est utile dans le cas où les prédictions sont mauvaises, car il possède un rapport de compétitivité $\max_i \frac{A(I)}{OPT(I)}$ de 2.

Notre implémentation se déroule en deux phases : l'initialisation (toutes les tâches démarrent) et l'exécution (*run*).

```
1     def _initRun(self):
2         for task in self.allTasks.values():
3             self.startTask(task)
4
5     def run(self, step):
6         if self.currentTime == 0:
7             self._initRun()
8
9         self.speed = self.initSpeed / len(self.workingTasks)
10        return self.work(step)
```

Il suffit de changer la vitesse de la machine en fonction du nombre de tâches restantes à chaque pas de temps.

3.4 Exécution parallèle

Comme vu précédemment, *Parallel* exécute *Prediction* et *Round-Robin* aux vitesses λ et $1 - \lambda$.

Ainsi, lors de l'initialisation de la machine, on définit ses sous-machines :

```
1         self.speed = speed
2         self.prediction = Prediction(speed * lmb, key)
3         self.roundRobin = RoundRobin(speed * (1-lmb), key)
```

Les deux machines possèdent les même tâches en référence, et travaillent ensemble à leur avancement.

La méthode *run* prend la forme suivante :

```

1  def run(self, step):
2      self.currentStep += step
3
4      self.finishTasks()
5
6      if not bool(self):
7          self.prediction.run(step)
8          self.roundRobin.run(step)
9
10     return bool(self)
11
12 def __bool__(self):
13     return bool(self.prediction) or bool(self.roundRobin)

```

Avec les méthodes *bool* renvoyant si les machines ont fini leur exécution ou non.

3.5 Exécution parallèle dynamique

Nous avons eu l'idée d'expérimenter une version de machine parallèle au λ dynamique.

Lorsqu'une tâche s'achève, on regarde à intervalle régulier la différence entre le temps d'exécution prédit, et le temps réellement effectué.

On utilise la formule suivante : $\lambda' = e^{-\epsilon_m}$ avec ϵ_m la moyenne de l'erreur observée.

```

1  def updateLmb(self):
2      if len(self.finishedTasks) != 0 and len(self.finishedTasks) % 5 == 0:
3          meanDiff = np.mean([abs(task.currentStep - task.predLength) for task
4                               ↪ in self.finishedTasks.values()])
5          newLmb = np.exp(-meanDiff*self.coef)
6
7          self.prediction.speed = self.speed * newLmb
8          self.roundRobin.initSpeed = self.speed * (1-newLmb)
9          self.historyLmb.append(newLmb)

```



```
9
10     def finishTasks(self):
11         super().finishTasks()
12         self.updateLmb()
```

On s'attend alors à observer un déplacement exponentiel du premier ordre des performances de la machine vers celles de Round-Robin.

Il s'agit de rajouter un coefficient α dans l'exponentielle pour adapter la vitesse de convergence de λ .

4 Test du bon fonctionnement des modèles

4.1 Modèle Prédiction

1

Machine 0	total : 3	working : 0	paused : 0	finished : 0	current time : 0		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 11	status : notAvailable		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : notAvailable		

À l'instant 0 toutes les tâches ne sont pas disponibles.

2

Machine 0	total : 3	working : 0	paused : 2	finished : 0	current time : 6		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 11	status : paused		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

Machine 0	total : 3	working : 1	paused : 1	finished : 0	current time : 7		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 10	status : working		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

À l'instant 6, la tâche 1 et 2 devienne disponibles, Prédiction réussi à sélectionner la tâche qui a la moins durée prédite.

3

Machine 0	total : 3	working : 1	paused : 2	finished : 0	current time : 16		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : paused		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 1	status : working		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

Machine 0	total : 3	working : 0	paused : 2	finished : 1	current time : 17		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : paused		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 0	status : finished		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

Machine 0	total : 3	working : 1	paused : 1	finished : 1	current time : 18		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 5	status : working		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 0	status : finished		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

À l'instant 17, la tâche 1 est terminée. Ensuite la tâche 0 a été sélectionnée.

4

Machine 0	total : 3	working : 1	paused : 0	finished : 2	current time : 41		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 0	status : finished		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 0	status : finished		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 1	status : working		

Machine 0	total : 3	working : 0	paused : 0	finished : 3	current time : 42		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 0	status : finished		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 0	status : finished		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 0	status : finished		

Toutes les tâches sont terminées à l'instant 42.

4.2 RR

Machine 2	total : 5	working : 0	paused : 0	finished : 0	current time : 5	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 7	status : notAvailable	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 7	status : notAvailable	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 8	status : notAvailable	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 19	status : notAvailable	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 28	status : notAvailable	
Machine 2	total : 5	working : 0	paused : 1	finished : 0	current time : 6	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 7	status : notAvailable	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 7	status : paused	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 8	status : notAvailable	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 19	status : notAvailable	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 28	status : notAvailable	
Machine 2	total : 5	working : 1	paused : 3	finished : 0	current time : 7	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 7	status : paused	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 6.0	status : working	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 8	status : paused	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 19	status : paused	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 28	status : notAvailable	

1

À l'instant 6, la tâche 7 a été sélectionné et allouée une vitesse de 1.

Machine 2	total : 5	working : 4	paused : 1	finished : 0	current time : 8	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 6.75	status : working	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 5.75	status : working	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 7.75	status : working	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 18.75	status : working	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 28	status : paused	
Machine 2	total : 5	working : 5	paused : 0	finished : 0	current time : 9	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 6.55	status : working	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 5.55	status : working	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 7.55	status : working	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 18.55	status : working	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 27.8	status : working	
Machine 2	total : 5	working : 5	paused : 0	finished : 0	current time : 10	
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : 6.35	status : working	
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : 5.35	status : working	
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : 7.35	status : working	
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : 18.35	status : working	
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 27.6	status : working	

2

À l'instant 9, toutes les tâches sont exécutées avec une vitesse 0.2.

Machine 2

total : 5

working : 5

paused : 0

finished : 0

current time : 36

Task 5

arrival time : 7

real length : 7

predicted length : 7

remaining : 1.15

status : working

Task 7

arrival time : 6

real length : 7

predicted length : 7

remaining : 0.15

status : working

Task 6

arrival time : 7

real length : 8

predicted length : 8

remaining : 2.15

status : working

Task 8

arrival time : 7

real length : 19

predicted length : 19

remaining : 13.15

status : working

Task 4

arrival time : 8

real length : 28

predicted length : 30

remaining : 22.4

status : working

Machine 2

total : 5

working : 4

paused : 0

finished : 1

current time : 37

Task 5

arrival time : 7

real length : 7

predicted length : 7

remaining : 0.95

status : working

Task 7

arrival time : 6

real length : 7

predicted length : 7

remaining : -0.05

status : finished

Task 6

arrival time : 7

real length : 8

predicted length : 8

remaining : 1.95

status : working

Task 8

arrival time : 7

real length : 19

predicted length : 19

remaining : 12.95

status : working

Task 4

arrival time : 8

real length : 28

predicted length : 30

remaining : 22.2

status : working

Machine 2

total : 5

working : 4

paused : 0

finished : 1

current time : 38

Task 5

arrival time : 7

real length : 7

predicted length : 7

remaining : 0.7

status : working

Task 7

arrival time : 6

real length : 7

predicted length : 7

remaining : -0.05

status : finished

Task 6

arrival time : 7

real length : 8

predicted length : 8

remaining : 1.7

status : working

Task 8

arrival time : 7

real length : 19

predicted length : 19

remaining : 12.7

status : working

Task 4

arrival time : 8

real length : 28

predicted length : 30

remaining : 21.95

status : working

3

À l'instant 37, la tâche 7 est terminée. Les tâches restantes ont une vitesse 0.25.

Machine 2	total : 5	working : 1	paused : 0	finished : 4	current time : 74		
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : -0.05	status : finished		
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : -0.05	status : finished		
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 1.2	status : working		
Machine 2	total : 5	working : 1	paused : 0	finished : 4	current time : 75		
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : -0.05	status : finished		
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : -0.05	status : finished		
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : 0.2	status : working		
Machine 2	total : 5	working : 0	paused : 0	finished : 5	current time : 76		
Task 5	arrival time : 7	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 7	arrival time : 6	real length : 7	predicted length : 7	remaining : -0.05	status : finished		
Task 6	arrival time : 7	real length : 8	predicted length : 8	remaining : -0.05	status : finished		
Task 8	arrival time : 7	real length : 19	predicted length : 19	remaining : -0.05	status : finished		
Task 4	arrival time : 8	real length : 28	predicted length : 30	remaining : -0.8	status : finished		

3

À l'instant 76 toutes les tâches sont terminées.

4.3 SPT

Machine 0	total : 3	working : 0	paused : 0	finished : 0	current time : 5		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 11	status : notAvailable		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : notAvailable		
Machine 0	total : 3	working : 0	paused : 2	finished : 0	current time : 6		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 11	status : paused		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		
Machine 0	total : 3	working : 1	paused : 1	finished : 0	current time : 7		
Task 0	arrival time : 8	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 6	real length : 11	predicted length : 12	remaining : 10	status : working		
Task 2	arrival time : 6	real length : 19	predicted length : 18	remaining : 19	status : paused		

1

À l'instant 6, la tâche 1 a été sélectionnée.

4.4 Parallèle

Parallèle 0	total : 3	current step : 1					
PredictionMachine 0	total : 3	working : 0	paused : 2	finished : 0	current time : 1		
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 11	status : paused		
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 19	status : paused		
speed : 0.5							
Round-RobinMachine 1	total : 3	working : 0	paused : 2	finished : 0	current time : 1		
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 11	status : paused		
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 19	status : paused		
speed : 0.5							
Parallèle 0	total : 3	current step : 2					
PredictionMachine 0	total : 3	working : 1	paused : 1	finished : 0	current time : 2		
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 10.25	status : working		
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.75	status : working		
speed : 0.5							
Round-RobinMachine 1	total : 3	working : 2	paused : 0	finished : 0	current time : 2		
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : notAvailable		
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 10.25	status : working		
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.75	status : working		
speed : 0.5							

1

À l'instant 1, la tâche 1 a été sélectionnée par Prédiction.

Donc à l'instant 2, la tâche 1 a été exécutée $0.5 \cdot 0.5 + 0.5 = 0.75$, alors que la tâche 2 a

$0.5 \times 0.5 = 0.25$ exécutée seulement par RR.

Parallel 0	total : 3	current step : 4				
PredictionMachine 0	total : 3	working : 1	paused : 2	finished : 0	current time : 4	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : paused	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 8.75	status : working	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.25	status : working	
speed : 0.5						
Round-RobinMachine 1	total : 3	working : 2	paused : 1	finished : 0	current time : 4	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 6	status : paused	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 8.75	status : working	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.25	status : working	
speed : 0.5						
Parallel 0	total : 3	current step : 5				
PredictionMachine 0	total : 3	working : 1	paused : 2	finished : 0	current time : 5	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 5.83	status : working	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 8.08	status : working	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.08	status : working	
speed : 0.5						
Round-RobinMachine 1	total : 3	working : 3	paused : 0	finished : 0	current time : 5	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : 5.83	status : working	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : 8.08	status : working	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 18.08	status : working	
speed : 0.5						

2

À l'instant 4, la tâche 0 est disponible. Le temps d'exécution par RR est $0.5 \times 0.33 = 0.165$.

Donc à l'instant 5, la tâche 1 a $8.75 - (0.165 + 0.5) = 8.085$ et les autres ont seulement un temps d'exécution 0.165.

Parallel 0	total : 3	current step : 24				
PredictionMachine 0	total : 3	working : 0	paused : 1	finished : 2	current time : 24	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : -0.58	status : finished	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : -0.58	status : finished	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 14.17	status : working	
speed : 0.5						
Round-RobinMachine 1	total : 3	working : 1	paused : 0	finished : 2	current time : 24	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : -0.58	status : finished	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : -0.58	status : finished	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 14.17	status : working	
speed : 0.5						
Parallel 0	total : 3	current step : 25				
PredictionMachine 0	total : 3	working : 1	paused : 0	finished : 2	current time : 25	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : -0.58	status : finished	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : -0.58	status : finished	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 13.17	status : working	
speed : 0.5						
Round-RobinMachine 1	total : 3	working : 1	paused : 0	finished : 2	current time : 25	
Task 0	arrival time : 4	real length : 6	predicted length : 5	remaining : -0.58	status : finished	
Task 1	arrival time : 1	real length : 11	predicted length : 12	remaining : -0.58	status : finished	
Task 2	arrival time : 1	real length : 19	predicted length : 18	remaining : 13.17	status : working	
speed : 0.5						

3

La tâche 2 a été allouée un temps d'exécution $= 0.5 + 0.5 \times 1 = 1$.

Le processus exécute jusqu'à la fin.

5 Résultats expérimentaux

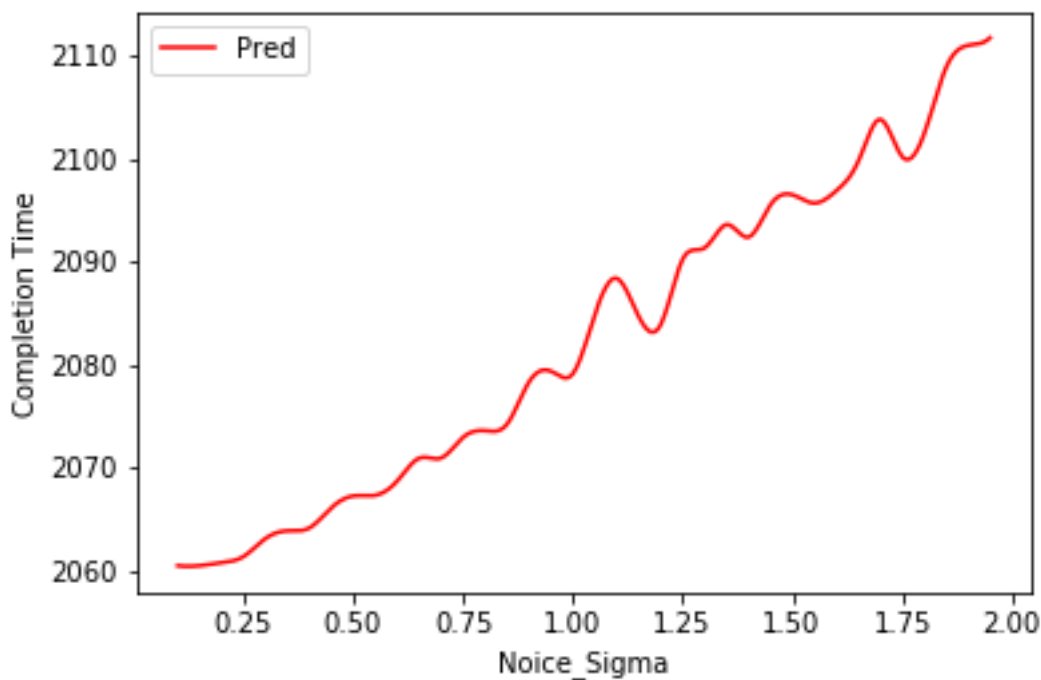
Pour effectuer les test, nous utilisons les distributions de probabilités suivantes:

- Distribution de Pareto pour la durée de tâche
- Distribution normale pour le bruit de la prédiction
- Distribution uniforme pour les dates d'arrivée des tâches. Ça indique que les tâches arrivent au fur et à mesure.

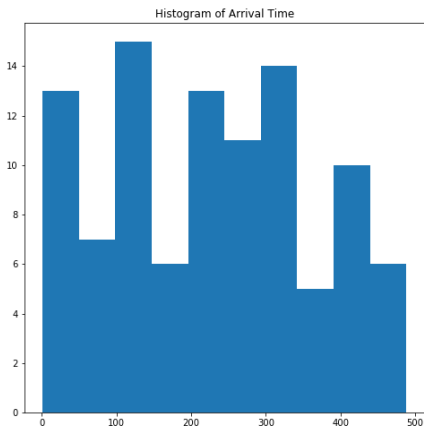
5.1 Statistiques des tâches

De plus, pour chaque algorithme nous tirons 100 tâches et observons les performances moyennes sur 100 exécutions différentes.

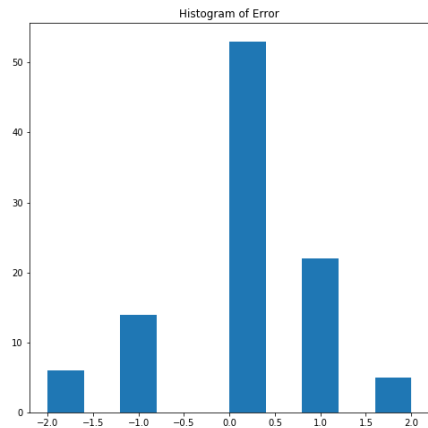
5.2 Introduction des dates d'arrivée



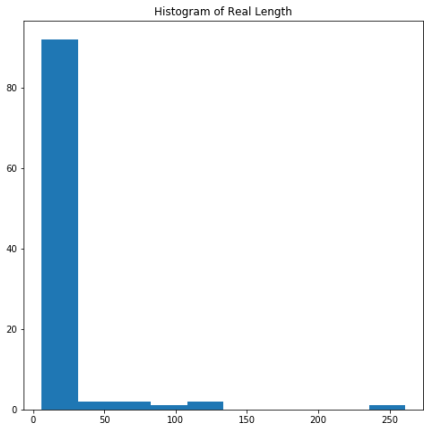
Nous pouvez visualiser très clairement que le temps de complétion augmente si la prédiction devient de plus en plus bruitée.



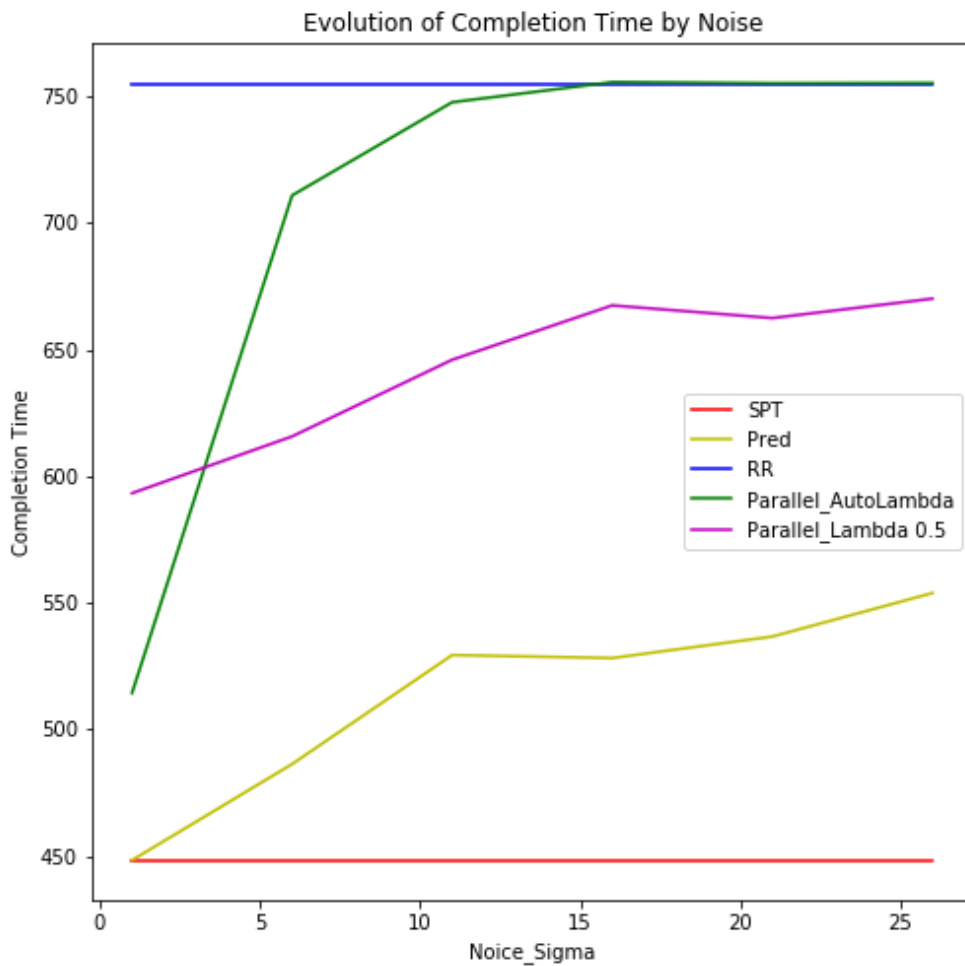
(a)



(b)



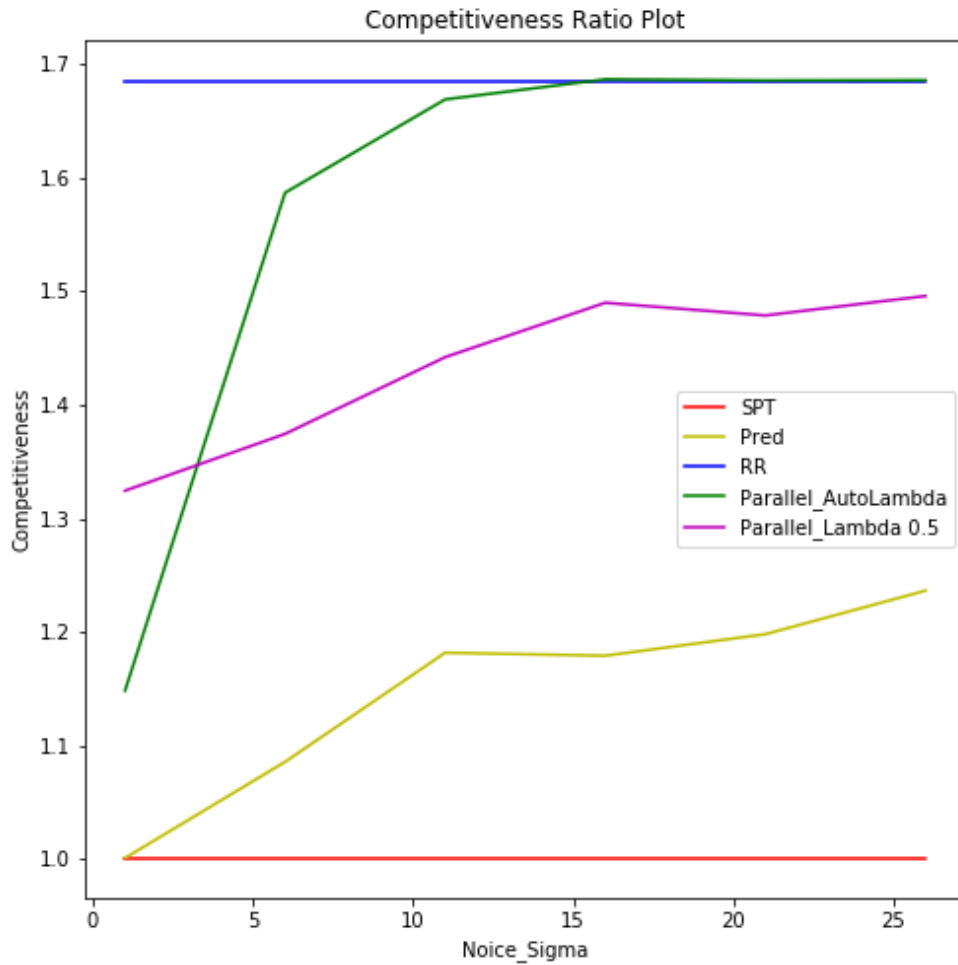
(c)



- 1 SPT devrait avoir de meilleures performances que tous les modèles.
- 2 Si le bruit des durées des tâches augmente, alors la performance de la Prédiction devient de plus en plus mauvaise. Parallel devrait rester constante, RR devrait rester constante aussi.
- 3 RR a un rapport de compétitivité de $2 * \text{SPT}$.
- 4 La machine Parallèle Auto-Lambda converge vers RR car moins la prédiction est bonne, plus on favorise RR.
- 5 Parallèle Avec Lambda=0.5 Fixé est un compromis entre Prediction et RR, sa performance

décroît avec celle de Prediction.

Les résultats expérimentaux sont cohérents avec nos hypothèses.



5.3 Cadre classique

Dans ce cadre-là, il s'agit d'annuler la date d'arrivée. Ensuite tout reste le même. Nous avons constaté que le temps de complétion est plus grand. C'est parce que nous avons la date d'arrivée où la tâche ne peut pas être exécutée.

