

TD de simulation : Modélisation

Exercice 1

Un bureau est composé de deux guichets en séries, chacun ayant une file d'attente gérée en FIFO. Un client ayant été servi au guichet 1 se rend au guichet 2 pour attendre son tour. On admet que les inter-arrivées des clients au guichet 1 sont iid et suivent des lois exponentielles d'espérance 1 minute, que les temps de service au guichet 1 sont des VAR exponentielle iid de paramètre 0,7 et que ceux au guichet 2 sont des VAR exponentielle iid de paramètre 0,9. On veut connaître, au bout de 1000 minutes, le temps d'attente espéré d'un client (dans les deux queues au global), la taille moyenne de la queue (au global) et le taux d'utilisation des guichets (indistinctement).

Écrire l'algorithme du simulateur

Comment change le modèle si on suppose qu'il y a un temps de déplacement entre le guichet 1 et la file du guichet 2? On supposera que ce temps de trajet, en minutes, suit la loi uniforme $\mathcal{U}[0, 2]$.

Comment change le modèle s'il n'y a pas de file d'attente possible au guichet 2 : un client qui vient d'être servi au guichet 1 et qui constate que le guichet 2 est occupé reste au guichet 1, bloquant ainsi les clients suivants?

Exercice 2

Une banque comporte m guichets ($m = 5$), chacun ayant sa propre file d'attente en FIFO. Quand un client arrive il choisit un guichet libre ou à défaut la file d'attente la plus courte. En cas d'égalité il prend le guichet ou la file le plus à droite. On note n_i le nombre de clients en face du guichet i (y compris celui qui est en train de se faire servir). Quand un client quitte un guichet i , si après mise à jour du n_i il existe une file j telle que $n_j > n_i + 1$ alors le dernier client de cette file saute dans la file i . En cas d'égalité c'est le client de la file la plus proche et, s'il y a toujours égalité, la plus à gauche qui change de file.

La banque ouvre à 8h00 et ferme à 5h00 mais continue à servir les derniers clients présents. Les inter-arrivées suivent une loi exponentielle d'espérance 1 min et les temps de service une loi exponentielle d'espérance 4,5 min. On veut connaître la durée espérée d'attente d'un client.

Ecrire l'algorithme du simulateur.

Exercice 3

Dans le problème de gestion de stock vu en cours, on suppose que si en début de mois le niveau de stock est négatif, l'entreprise peut demander une réappro d'urgence. Dans ce cas la livraison se fait dans un délai de loi $\mathcal{U}[0, 25; 0, 5]$ avec un coût de $48 + 4Z$ euros (Z étant la quantité demandée)

modifier en conséquence l'algo de simulation

Exercice 4

Les bus d'une cité arrivent à un centre de maintenance avec des inter-arrivées exponentielles de moyenne 2 heures. Le centre est composé d'un poste de contrôle et de deux postes de réparations. Chaque bus est contrôlé (durée uniforme entre 15min et 1h05) et si un défaut est constaté il passe dans la file d'attente de réparation (une file unique pour les deux postes, gérée en FIFO). La durée de la réparation suit une loi uniforme entre 2,1 et 4,5 heures.

Un bus a 30% de chance de devoir subir une réparation. On veut effectuer une simulation sur 160 heures et connaître le temps moyen d'attente dans chaque queue et le taux d'utilisation du centre de réparation.

Exercice 1

Deux guichets en série - 1er cas - Graphe d'événements détaillé.

Procédure ArrivéeClient

1. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$
2. Incrémenter le nombre de clients ($NbClients \leftarrow NbClients + 1$)
3. Insérer dans l'échéancier un événement *ArrivéeFile1* à l'instant $DateSimu$

Procédure ArrivéeFile1

1. Incrémenter le nombre de clients dans la File 1 ($Q_1 \leftarrow Q_1 + 1$)
2. **Si** le Guichet 1 est libre (i.e. $B_1 = 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet1* à l'instant $DateSimu$
4. **Fin Si**

Procédure AccèsGuichet1

1. Décrémenter le nombre de clients dans la File 1 ($Q_1 \leftarrow Q_1 - 1$)
2. Changer le statut du Guichet 1 en "occupé" ($B_1 \leftarrow 1$)
3. Insérer dans l'échéancier un événement *DépartGuichet1* à l'instant $DateSimu + \mathcal{E}(0,7)$

Procédure DépartGuichet1

1. Changer le statut du Guichet 1 en "libre" ($B_1 \leftarrow 0$)
2. **Si** la File 1 est non vide (i.e. $Q_1 > 0$) **alors**
3. Insérer dans l'échéancier un événement de *AccèsGuichet1* à l'instant $DateSimu$
4. **Fin Si**
5. Insérer dans l'échéancier un événement de *ArrivéeFile2* à l'instant $DateSimu$

Procédure ArrivéeFile2

1. Incrémenter le nombre de clients dans la File 2 ($Q_2 \leftarrow Q_2 + 1$)
2. **Si** le Guichet 2 est libre (i.e. $B_2 = 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet2* à l'instant $DateSimu$
4. **Fin Si**

Procédure AccèsGuichet2

1. Décrémenter le nombre de clients dans la File 2 ($Q_2 \leftarrow Q_2 - 1$)
2. Changer le statut du Guichet 2 en "occupé" ($B_2 \leftarrow 1$)
3. Insérer dans l'échéancier un événement *DépartGuichet2* à l'instant $DateSimu + \mathcal{E}(0,9)$

Procédure DépartGuichet2

1. Changer le statut du Guichet 2 en "libre" ($B_2 \leftarrow 0$)
2. **Si** la File 2 est non vide (i.e. $Q_2 > 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet2* à l'instant $DateSimu$
4. **Fin Si**

Procédure DébutSimulation

1. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$
2. Insérer dans l'échéancier un événement *FinSimulation* à l'instant $DateSimu + 1000$
3. //Initialiser les variables
4. $Q_1 \leftarrow 0; Q_2 \leftarrow 0$

5. $B_1 \leftarrow 0; B_2 \leftarrow 0$
6. //Initialiser les indicateurs
7. $AireQ_1 \leftarrow 0; AireQ_2 \leftarrow 0$
8. $AireB_1 \leftarrow 0; AireB_2 \leftarrow 0$
9. $NbClients \leftarrow 0$

Procédure FinSimulation

1. Vider l'échéancier
2. // Calculer les indicateurs
3. $Temps\ d'attente\ moyen\ d'un\ client \leftarrow (AireQ_1 + AireQ_2)/NbClients$
4. $Nombre\ moyen\ de\ clients\ en\ attente \leftarrow (AireQ_1 + AireQ_2)/1000$
5. $Taux\ moyen\ d'utilisation\ des\ guichets \leftarrow (AireB_1 + AireB_2)/2000$

Procédure Simulateur

1. $DateSimu \leftarrow 480$ (i.e. 8h00)
2. Insérer dans l'échéancier un événement *DébutSimulation* à la date $DateSimu$
3. **Tant que** l'échéancier n'est pas vide **Faire**
4. Extraire le premier couple $(Evt, Date)$ de l'échéancier
5. Mise-à-JourDesAires($DateSimu$, $Date$)
6. $DateSimu \leftarrow Date$
7. Exécuter l'événement Evt
8. **Fin Tant que**

Procédure Mise-à-JourDesAires(D1,D2)

1. $AireQ_1 \leftarrow AireQ_1 + (D2-D1) \cdot Q_1$
2. $AireB_1 \leftarrow AireB_1 + (D2-D1) \cdot B_1$
3. $AireQ_2 \leftarrow AireQ_2 + (D2-D1) \cdot Q_2$
4. $AireB_2 \leftarrow AireB_2 + (D2-D1) \cdot B_2$

Deux guichets en série - 3ième cas - Graphe d'événements détaillé.

Procédure ArrivéeClient

1. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$
2. Incrémenter le nombre de clients ($NbClients \leftarrow NbClients + 1$)
3. Insérer dans l'échéancier un événement *ArrivéeFile1* à l'instant $DateSimu$

Procédure ArrivéeFile1

1. Incrémenter le nombre de clients dans la File 1 ($Q_1 \leftarrow Q_1 + 1$)
2. **Si** le Guichet 1 est libre (i.e. $B_1 = Bloq_1 = 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet1* à l'instant $DateSimu$
4. **Fin Si**

Procédure AccèsGuichet1

1. Décrémenter le nombre de clients dans la File 1 ($Q_1 \leftarrow Q_1 - 1$)
2. Changer le statut du Guichet 1 en "occupé" ($B_1 \leftarrow 1$)
3. Insérer dans l'échéancier un événement *FinServiceGuichet1* à l'instant $DateSimu + \mathcal{E}(0,7)$

Procédure FinServiceGuichet1

1. Changer le statut du Guichet 1 en "libre" ($B_1 \leftarrow 0$)
2. Changer le statut du Guichet 1 en "bloqué" ($Bloq_1 \leftarrow 1$)
3. **Si** le Guichet 2 est libre (i.e. $B_2 = 0$) **alors**
4. Insérer dans l'échéancier un événement *DépartGuichet1* à l'instant $DateSimu$

Procédure DépartGuichet1

1. Changer le statut du Guichet 1 en "non bloqué" ($Bloq_1 \leftarrow 0$)
2. Insérer dans l'échéancier un événement *AccèsGuichet2* à l'instant $DateSimu$
3. **Si** la File 1 est non vide (i.e. $Q_1 > 0$) **alors**
4. Insérer dans l'échéancier un événement *AccèsGuichet1* à l'instant $DateSimu$
5. **Fin Si**

Procédure AccèsGuichet2

1. Changer le statut du Guichet 2 en "occupé" ($B_2 \leftarrow 1$)
2. Insérer dans l'échéancier un événement *DépartGuichet2* à l'instant $DateSimu + \mathcal{E}(0,9)$

Procédure DépartGuichet2

1. Changer le statut du Guichet 2 en "libre" ($B_2 \leftarrow 0$)
2. **Si** le Guichet 1 est bloqué (i.e. $Bloc_1 = 1$) **alors**
3. Insérer dans l'échéancier un événement *DépartGuichet1* à l'instant $DateSimu$
4. **Fin Si**

Procédure DébutSimulation

1. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$
2. Insérer dans l'échéancier un événement *FinSimulation* à l'instant $DateSimu + 1000$
3. $Q_1 \leftarrow 0$; $Bloq_1 \leftarrow 0$
4. $B_1 \leftarrow 0$; $B_2 \leftarrow 0$
5. $AireQ_1 \leftarrow 0$; $AireBloq_1 \leftarrow 0$
6. $AireB_1 \leftarrow 0$; $AireB_2 \leftarrow 0$

7. $NbClients \leftarrow 0$

Procédure FinSimulation

1. Vider l'échéancier
2. // Calculer les indicateurs
3. $Temps\ d'attente\ moyen\ d'un\ client \leftarrow (AireQ_1 + AireBloq_1)/NbClients$
4. $Nombre\ moyen\ de\ clients\ en\ attente \leftarrow (AireQ_1 + AireBloq_1)/1000$
5. $Taux\ moyen\ d'utilisation\ des\ guichets \leftarrow (AireB_1 + AireB_2)/2000$

Procédure Simulateur

1. $DateSimu \leftarrow 480$ (i.e. 8h00)
2. Insérer dans l'échéancier un événement *DébutSimulation* à la date $DateSimu$
3. **Tant que** l'échéancier n'est pas vide **Faire**
4. Extraire le premier couple $(Evt, Date)$ de l'échéancier
5. Mise-à-JourDesAires($DateSimu$, $Date$)
6. $DateSimu \leftarrow Date$
7. Exécuter l'événement Evt
8. **Fin Tant que**

Procédure Mise-à-JourDesAires(D1,D2)

1. $AireQ_1 \leftarrow AireQ_1 + (D2-D1) \cdot Q_1$
2. $AireB_1 \leftarrow AireB_1 + (D2-D1) \cdot B_1$
3. $AireBloc_1 \leftarrow AireBloq_1 + (D2-D1) \cdot Bloq_1$
4. $AireB_2 \leftarrow AireB_2 + (D2-D1) \cdot B_2$

Banque 5 guichets

Procédure ArrivéeClient

1. **Si** $DateSimu < 17h00$ **alors**
2. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$
3. Incrémenter le nombre de clients ($NbClients \leftarrow NbClients + 1$)
4. Calculer la taille de la file la plus courte ($Q \leftarrow \min_{1 \leq i \leq 5} \{Q_i + B_i\}$)
5. Identifier la file la plus à droite ($k \leftarrow \min\{i \mid Q_i + B_i = Q\}$)
6. Insérer dans l'échéancier un événement *ArrivéeFile-k* à l'instant $DateSimu$
7. **Fin Si**

Procédure ArrivéeFile-i

1. Incrémenter le nombre de clients dans la File i ($Q_i \leftarrow Q_i + 1$)
2. **Si** le Guichet i est libre (i.e. $B_i = 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet-i* à l'instant $DateSimu$
4. **Fin Si**

Procédure AccèsGuichet-i

1. Décrémenter le nombre de clients dans la File i ($Q_i \leftarrow Q_i - 1$)
2. Changer le statut du Guichet i en "occupé" ($B_i \leftarrow 1$)
3. Insérer dans l'échéancier un événement *DépartGuichet-i* à l'instant $DateSimu + \mathcal{E}(1/4.5)$

Procédure DépartGuichet-i

1. Changer le statut du Guichet i en "libre" ($B_i \leftarrow 0$)
2. **Si** la File i est non vide (i.e. $Q_i > 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsGuichet-i* à l'instant $DateSimu$
4. **Fin Si**
5. Insérer dans l'échéancier un événement *Réorg-i* à l'instant $DateSimu$
6. **Si** $DateSimu \geq 17h00$ et $\max_{1 \leq i \leq 5} (Q_i + B_i) = 0$ **alors**
7. Insérer dans l'échéancier un événement *Fin* à l'instant $DateSimu$

Procédure Réorg-i

1. $j_1 \leftarrow \max\{k < i \mid Q_k + B_k > Q_i + B_i + 1\}$ (si une telle File n'existe pas alors $j_1 \leftarrow -\infty$)
2. // j_1 désigne la file candidate la plus proche à droite de la file i.
3. $j_2 \leftarrow \min\{k > i \mid Q_k + B_k > Q_i + B_i + 1\}$ (si une telle File n'existe pas alors $j_2 \leftarrow +\infty$)
4. // j_2 désigne la file candidate la plus proche à gauche de la file i.
5. $j \leftarrow j_2$ // On choisit la file de gauche
6. // Mais si la file de droite est strictement plus proche alors on la préfère.
7. **Si** $|j_1 - i| < |j_2 - i|$
8. **alors** $j \leftarrow j_1$
9. **Fin Si**
10. **Si** $|j| < \infty$ **alors** (// S'il existe une file candidate)
11. Décrémenter le nombre de clients dans la File j ($Q_j \leftarrow Q_j - 1$)
12. Insérer dans l'échéancier un événement *ArrivéeFile-i* à l'instant $DateSimu$
13. **Fin Si**

Procédure Début Simulation

1. $NbClients \leftarrow 0$

2. $\forall i \in \llbracket 1, 5 \rrbracket : Q_i \leftarrow 0$
3. $\forall i \in \llbracket 1, 5 \rrbracket : AireB_i \leftarrow 0$
4. $\forall i \in \llbracket 1, 5 \rrbracket : AireQ_i \leftarrow 0$
5. Insérer dans l'échéancier un événement *ArrivéeClient* à l'instant $DateSimu + \mathcal{E}(1)$

Procédure Fin Simulation

1. Vider l'échéancier
2. $Temps\ d'attente\ moyen\ d'un\ client \leftarrow \sum_i (AireQ_i) / NbClients$

Procédure Mise-à-JourDesAires(D1,D2)

1. $\forall i \in \llbracket 1, 5 \rrbracket : AireQ_i \leftarrow AireQ_i + (D2 - D1) \cdot Q_i$

Procédure Simulateur

1. $DateSimu \leftarrow 08h00$
2. Insérer dans l'échéancier un événement Début à l'instant $DateSimu$
3. **Tant que** l'échéancier n'est pas vide **Faire**
4. Extraire le premier couple $(Evt, Date)$ de l'échéancier
5. Mise-à-JourDesAires($DateSimu$, $Date$)
6. $DateSimu \leftarrow Date$
7. Exécuter l'événement Evt
8. **Fin Tant que**

Centre de maintenance de bus

Procédure ArrivéeBus

1. Insérer dans l'échéancier un événement *ArrivéeBus* à l'instant $DateSimu + \mathcal{E}(1/2)$
2. Incrémenter le nombre de bus ($NbBus \leftarrow NbBus + 1$)
3. Insérer dans l'échéancier un événement *ArrivéeFileC* à l'instant $DateSimu$

Procédure ArrivéeFileC

1. Incrémenter le nombre de bus dans la File C ($Q_C \leftarrow Q_C + 1$)
2. **Si** le centre de contrôle est libre (i.e. $B_C = 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsContrôle* à l'instant $DateSimu$
4. **Fin Si**

Procédure AccèsContrôle

1. Décrémenter le nombre de bus dans la File C ($Q_C \leftarrow Q_C - 1$)
2. Changer le statut centre de contrôle en "occupé" ($B_C \leftarrow 1$)
3. Insérer dans l'échéancier un événement *DépartContrôle* à l'instant $DateSimu + \mathcal{U}([1/4, 13/12])$

Procédure DépartContrôle

1. Changer le statut du centre de contrôle en "libre" ($B_C \leftarrow 0$)
2. **Si** la File C est non vide (i.e. $Q_C > 0$) **alors**
3. Insérer dans l'échéancier un événement de *AccèsContrôle* à l'instant $DateSimu$
4. **Fin Si**
5. **Si** random < 30% **alors** (*// Il y a une probabilité de 30% que le bus subisse une réparation*)
6. Insérer dans l'échéancier un événement *ArrivéeFileR* à l'instant $DateSimu$
7. **Fin Si**

Procédure ArrivéeFileR

1. Incrémenter le nombre de bus dans la File R ($Q_R \leftarrow Q_R + 1$)
2. Incrémenter le nombre de bus réparés ($NbBusRep \leftarrow NbBusRep + 1$)
3. **Si** le centre de réparation a de la place (i.e. $B_R < 2$) **alors**
4. Insérer dans l'échéancier un événement *AccèsRéparation* à l'instant $DateSimu$
5. **Fin Si**

Procédure AccèsRéparation

1. Décrémenter le nombre de bus dans la File R ($Q_R \leftarrow Q_R - 1$)
2. Réquisitionner un poste dans le centre de réparation ($B_R \leftarrow B_R + 1$)
3. Insérer dans l'échéancier un événement *DépartRéparation* à l'instant $DateSimu + \mathcal{U}([2.1, 4.5])$

Procédure DépartRéparation

1. Libérer un poste du centre de réparation ($B_R \leftarrow B_R - 1$)
2. **Si** la File R est non vide (i.e. $Q_R > 0$) **alors**
3. Insérer dans l'échéancier un événement *AccèsRéparation* à l'instant $DateSimu$
4. **Fin Si**

Procédure Début Simulation

1. $NbBus \leftarrow 0$; $NbBusRep \leftarrow 0$
2. $AireQ_C \leftarrow 0$; $AireQ_R \leftarrow 0$; $AireB_R \leftarrow 0$
3. $Q_C \leftarrow 0$; $Q_R \leftarrow 0$; $B_C \leftarrow 0$; $B_R \leftarrow 0$

4. Insérer dans l'échéancier un événement *ArrivéeBus* à l'instant $DateSimu + \mathcal{E}(1/2)$
5. Insérer dans l'échéancier un événement *Fin* à l'instant 160

Procédure Fin Simulation

1. Vider l'échéancier
2. $Temps\ d'attente\ moyen\ avant\ contrôle \leftarrow AireQ_C / NbBus$
3. $Temps\ d'attente\ moyen\ avant\ réparation \leftarrow AireQ_R / NbBusRep$
4. $Taux\ d'utilisation\ du\ centre\ de\ réparation \leftarrow AireB_R / (2 \times 160)$

Procédure Mise-à-JourDesAires(D1,D2)

1. $AireQ_C \leftarrow AireQ_C + (D2 - D1) \cdot Q_C$
2. $AireQ_R \leftarrow AireQ_R + (D2 - D1) \cdot Q_R$
3. $AireB_R \leftarrow AireB_R + (D2 - D1) \cdot B_R$

Procédure Simulateur

1. $DateSimu \leftarrow 00h00$
2. Insérer dans l'échéancier un événement *Début* à l'instant $DateSimu$
3. **Tant que** l'échéancier n'est pas vide **Faire**
4. Extraire le premier couple $(Evt, Date)$ de l'échéancier
5. Mise-à-JourDesAires($DateSimu$, $Date$)
6. $DateSimu \leftarrow Date$
7. Exécuter l'événement Evt
8. **Fin Tant que**