

# Simulation à événements discrets

## Introduction

### Objectifs

- Développer un schème commun de la modélisation des systèmes complexes
- Compatible aux blocs basiques dans les modèles des événements discrets

### Systèmes étudiés:

- Stochastiques
- Dynamiques

### Systèmes/modèles à événements discrets

- Systèmes dont les états changent seulement dans aux moments discrets

## Définitions

### Système

- Une collection des entités qui interagissent entre eux au fil du temps
- Exemple:

Système = Machine(s) + Utilisateurs

### Modèle

- Une représentation abstraite d'un système
- Exemple:
  - Fils d'attente
  - Réseaux de fils d'attente

## Définitions

### État de système

- Une collection des variables contenant toutes les informations nécessaires à décrire un système à tout moment
- Exemple:
  - Réseaux de fils d'attente:
    - Nombre de clients servis dans le fils d'attente
    - Nombre de clients en attente dans le fils d'attente
    - Nombre de machines libres
    - Nombre de machines occupées
  - .....

## Définitions

### Entité

- Un objet ou d'un composant dans le système
- Exemple:
  - un serveur:
    - Souvent avec une capacité limitée
    - réparateur, machine, piste d'aéroport,...
  - un client:
    - Associé à un service spécifique demandé
    - utilisateur, camion, emails,...
  - ...

## Définitions

### Attribut

- Les propriétés d'un entité donné
- Exemple:
  - Serveur
    - Capacité de service
    - Nombre maximal des clients en attente
    - Horaire d'ouverture
    - Temps de service nécessaire d'un client
    - Stratégie d'ordonnance des tâches/clients dans un fils d'attente

Définitions

Attribut

- Les propriétés d'un entité donné
- Exemple:
  - Client
    - Temps d'arrivée
    - Tâche/service demandé au serveur
      - Durée
      - Ressource demandée
    - Durée maximale d'attente

Définitions

Listes

- Une collection des entités associés ordonnés selon une certaine opération logique
- Exemple:
  - Ensembles: sans réplication
  - Fils d'attente: selon la stratégie du service souvent le temps d'arrivée
  - Chaînes

Définitions

Événements

- Un événement instantané qui change l'état d'un système
- Exemple:
  - Arrivée d'un nouveau client: Nombre de clients dans le fils d'attente +
  - Départ d'un client: Nombre de clients dans le fils d'attente - 1

Définitions

Liste des événements:

- Une liste des notifications d'événements pour les événements futurs, ordonnés par l'instant d'arrivée de l'événement,
- Aussi appelée Liste des événements à la future (LEF).
- Exemple
  - Listes des clients à arriver
    - Temps d'arrivée
    - Service demandé
  - Durée et ressource nécessaires pour les tâches demandés

Définitions

Activité:

- Une durée de temps spécifiée avec l'instant de début connu
- Exemple:
  - Durée de service.
  - Entre-temps d'arrivée des clients

Définitions

Horloge:

- Une variable représentant le temps simulé.
  - Valeur maximale souvent fixée comme critère d'arrêt de simulation
- Remarque

## Activité

Une activité représente un temps de service, un temps d'inter-arrivée, ou tout temps de traitement dont la durée a été caractérisée/définie par le modèle:

La durée d'une activité peut être spécifiée:

- Déterministe
- Statistique.
- Une fonction dépendant des variables système et/ou des attributs d'entité (Ressource).

La durée n'est pas affectée par l'occurrence d'autres événements, par conséquent, l'activité est également appelée une attente inconditionnelle.

## Activité

Réalisation d'une activité est un événement, souvent appelé un événement primaire.

Exemple

Si l'heure simulée actuelle est

$$\text{Horloge } t = 100 \text{ minutes}$$

et qu'un temps d'inspection/service 'de  $ct = 5$  minutes commence à peine, une notification d'événement est créée qui spécifie le type d'événement et l'heure de l'événement

$$t' = t + \delta t$$

$$100 + 5 = 105 \text{ minutes}$$

## Activité

La durée d'un délai est déterminée par les conditions du système

- Non spécifiées par le modèle à l'avance
- Aussi appelé une attente conditionnelle.

Exemple

- Le retard d'un client dans une file d'attente peut dépendre du nombre et de la durée de service d'autres clients en place et de la défaillance d'un serveur pendant le délai.

## Dynamique

Dynamique:

- En fonction du temps et en constante évolution dans le temps
- L'état du système, les attributs de l'entité, le nombre d'entités actives, le contenu des ensembles, les activités et les retards actuellement en cours sont toutes des fonctions du temps.

## Dynamique

La définition des composants du modèle fournit une description statique du modèle.

La description des relations dynamiques et des interactions entre les composants est également nécessaire.

Exemples:

- Comment chaque événement affecte-t-il l'état du système?
- Quels événements marquent le début ou la fin de chaque activité?
- Quel est l'état du système à l'instant 0?

## Dynamique

Une simulation à événement discret est:

- La modélisation dans le temps d'un système dont tous les changements d'état se produisent à des moments discrets.
- Procède en produisant une séquence d'instantanés système.

## Algorithme de planification d'événements/d'avancement de temps

Le mécanisme pour avancer le temps de simulation et garantir que tous les événements à se produire dans l'ordre chronologique correctement.

À tout instant  $t$ , la liste d'événements futurs (LEF) contient tous les événements futurs programmés précédemment et leurs temps d'événements associés  $(t_1, t_2, \dots)$ :

- LEF ordonnée par temps d'événement croissante
- Le temps d'événement satisfait

$$t \leq t_1 \leq t_2 \leq \dots \leq t_n$$

où  $t$  est la valeur de l'Horloge  $t$ .

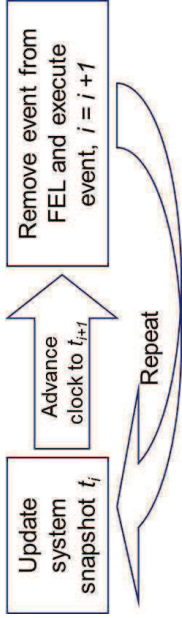
## Algorithme de planification d'événements/d'avance de temps

Système à l'instant  $t_1$

- Retirer l'événement  $(1, t_1)$  de la liste
- Avancer l'horloge  $t_0 \rightarrow t_1$
- Exécuter la tâche  $(1, t_1)$ :
  - Mettre à jours les états du systèmes
  - Changer les attributs des entités
  - ....

Horloge	État du systèmes	Liste des événements futures (LEF)
$t_1$	(10,2,7)	(1,2) (1,3) ... (2,n)

## Algorithme de planification d'événements/d'avance de temps



État du système à l'instant  $t_0$

- Événements à arriver  $(i, t)$ 
  - Type d'événement:  $i$
  - Temps d'arrivée:  $t$
- Liste des événements futures ordonnée à l'ordre croissante des temps d'arrivée

Horloge	État du systèmes	Liste des événements futures (LEF)
$t_0$	(5,1,6)	(3,1) Type 3 à $t_1$ (1,2) Type 1 à $t_2$ (1,3) Type 1 à $t_3$ ... (2,n) Type 1 à $t_n$

## Traitement de liste

La gestion d'une liste:

- Les principales opérations de traitement de liste effectuées sur une LEF sont:
  - Retrait de l'événement imminent
  - Ajout d'un nouvel événement à la liste.
  - De temps en temps l'enlèvement d'un événement (annulation d'un événement).
- L'efficacité de la recherche dans la liste dépend de l'organisation logique de la liste et de la manière dont la recherche est effectuée.

## Algorithme de planification d'événements/d'avance de temps

Système à l'instant  $t_1$

- Générer ou trouver le prochain événement à arriver selon leur temps d'arrivée:
  - Ex: Événement  $(4, t^*)$  Type 4 à  $t^*$   
 $t_2 < t^* < t_3$
- Mettre à jour les statistiques cumulatives

Horloge	État du systèmes	Liste des événements futures (LEF)
$t_1$	(10,2,7)	(1,2) (4,t*) (1,3) ... (2,n)

## Traitement de liste

Lorsqu'un événement avec un temps d'événement est généré, sa position correcte sur la LEF peut être trouvée via:

- Une recherche descendante
- Une recherche ascendante.

## Événements à venir

Un événement exogène est un événement «hors du système» qui affecte le système

- Exemple:  
l'arrivée et l'achèvement du service dans un système de mise en file d'attente.

Événement d'arrivée:

- Exemple:  
une arrivée à un système de file d'attente, à l'instant 0, le premier événement d'arrivée est généré et est programmé sur la LEF.

Lorsque l'horloge est finalement avancée jusqu'au moment de cette première arrivée, un deuxième événement d'arrivée est généré.

## Événements à venir

Événement d'arrivée:

- La fin d'un intervalle entre les arrivées est un exemple d'événement primaire (l'événement principal est géré en plaçant un avis d'événement sur le LEF).

Événement de fin de service:

- Événement conditionnel.
- Déclenchée uniquement à la condition qu'un client soit présent et un serveur est libre.
- Un temps de service est un exemple d'activité.

## Événements à venir

Alterner des générations de temps d'exécution et d'arrêt pour une machine soumise à des pannes.

Événement d'arrêt,  $E^-$ :

- A l'instant 0 programmer un événement de simulation d'arrêt à un instant futur spécifié  $T_E^-$ .
- La longueur de parcours  $T_E^-$  est déterminée par la simulation elle-même.
- Généralement,  $T_E^-$  est le moment de l'occurrence d'un événement spécifié  $E^-$ .

## Vues du monde

Les vues du monde les plus répandues sont:

- Vue du monde d'e planification d'événements
  - Avance de temps variable
- Vue du monde d'interaction de processus
  - Avance temporelle variable
- Vue du monde d'analyse d'activité
  - Incrément de temps fixe

## Vues du monde

Approche de planification d'événements:

- Se concentre sur les événements et leurs effets sur l'état du système.
- Résumée dans les diapositives précédentes

## Vues du monde

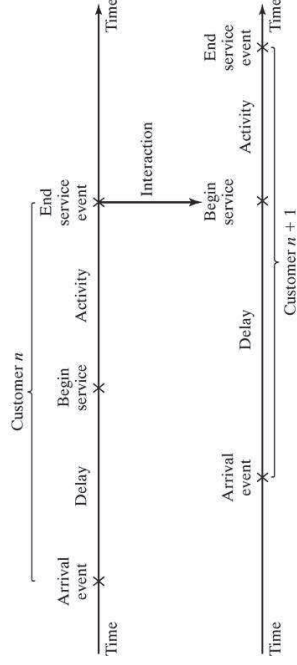
Approche d'interaction de processus:

- Se concentre sur les processus: un processus est une liste chronologique des événements, des activités et des retards qui définissent le cycle de vie d'une entité lorsqu'elle se déplace dans un système.
- Habituellement, de nombreux processus sont actifs simultanément dans un modèle, et l'interaction entre les processus peut être assez complexe.
- Une approche populaire car elle a un attrait intuitif et facilite la mise en œuvre du logiciel de simulation.

## Vues du monde

Approche d'interaction de processus:

- Exemple: Processus client



## Vues du monde

L'approche modifiée est appelée l'approche en trois phases.

- Les événements sont considérés comme des activités de durée zéro unités de temps.
- Les activités sont divisées en 2 catégories:
  - Type B: activités qui sont susceptibles de se produire, par exemple, des événements primaires.
  - Type C: activités ou événements qui sont conditionnels à ce que certaines conditions soient vraies.
- La simulation continue avec l'exécution répétée des 3 phases: à partir de la suppression des événements, puis exécution des événements de type B, puis exécution des événements de type C.

## Vues du monde

Approche d'analyse d'activité:

- Se concentre sur les activités d'un modèle et les conditions qui permettent à une activité de commencer.
- A chaque avance d'horloge, les conditions de chaque activité sont vérifiées, et si les conditions sont vraies, alors les activités correspondantes commencent.
- Simple dans le concept, mais l'exécution lente sur les ordinateurs.

## Ciw

Ciw est une bibliothèque de simulation d'événements discrets pour les réseaux de files d'attente ouverts.

- Ses principales caractéristiques comprennent
  - la capacité de simuler des réseaux de files d'attente, plusieurs classes de clients et
  - la mise en œuvre du blocage de type I pour les réseaux restreints.
- Un certain nombre d'autres fonctionnalités sont également mises en œuvre, y compris
  - les priorités,
  - le calage,
  - les planifications et
  - la détection d'interblocage.

## Ciw

## Introduction

```
Importer Ciw
import ciw

Objet réseau de fils d'attente: Network
• Nombre de serveurs
• Distribution de temps d'entres-arrivée
• Distribution de temps de service
N = ciw.create_network(
    Arrival_distributions=[['Exponential', 0.2]],
    Service_distributions=[['Exponential', 0.1]],
    Number_of_servers=3
)
```

## Fils d'attente

```
>>> import ciw
>>> N = ciw.create_network(
...     Arrival_distributions=[['Exponential', 5.0]],
...     Service_distributions=[['Exponential', 8.0]],
...     Transition_matrices=[[0.0]],
...     Number_of_servers=1
... )
```

## Fils d'attente

```
average_waits = []
warmup = 10
cooldown = 10
maxsimtime = 40
```

## Fils d'attente

```
for s in range(25):
    ciw.seed(s)
    Q = ciw.Simulation(N)
    Q.simulate_until_max_time(warmup + maxsimtime + cooldown)
    recs = Q.get_all_records()
    waits = [r.waiting_time for r in recs if r.arrival_date > warmup
and r.arrival_date < warmup + maxsimtime]
    average_waits.append(sum(waits) / len(waits))
```

## Fils d'attente

```
average_wait = sum(average_waits) / len(average_waits)
average_wait
```

## Réseau de fils d'attente

```
>>> import ciw
>>> N = ciw.create_network(
...     Arrival_distributions=[['Exponential', 0.3],
...     ['Exponential', 0.2],
...     'NoArrivals'],
...     Service_distributions=[['Exponential', 1.0],
...     ['Exponential', 0.4],
...     ['Exponential', 0.5]],
...     Transition_matrices=[[0.0, 0.3, 0.7],
...     [0.0, 0.0, 1.0],
...     [0.0, 0.0, 0.0]],
...     Number_of_servers=[1, 2, 2]
... )
```

## Réseau de fils d'attente

```
>>> completed_custs = []
>>> for trial in range(10):
...     ciw.seed(trial)
...     Q = ciw.Simulation(N)
...     Q.simulate_until_max_time(200)
...     recs = Q.get_all_records()
...     num_completed = len([r for r in recs if r.node==3 and r.arrival_date
... < 180])
...     completed_custs.append(num_completed)
>>> sum(completed_custs) / len(completed_custs)
```

## Exemple d'épicerie

Composants du modèle:

- Etat du système:
  - LQ (t) - Nombre de clients en attente à l'instant t,
  - LS (t) - Nombre de clients étant servis à l'instant t.
- Entités:
  - Le serveur et les clients ne sont pas explicitement modélisés, sauf en termes de variables d'état.
- Événements:
  - Arrivée (A),
  - Départ (D),
  - Événement d'arrêt (E).

## Approche de planification des événements

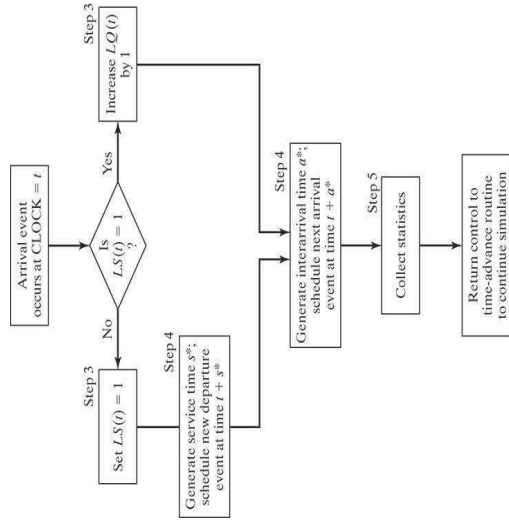
Exemple d'épicerie

## Exemple d'épicerie

Exemple d'épicerie :

File d'attente à un seul serveur (caisse unique).

- Système = caisse unique + clients
- Nombre de clients dans le système= nombre de clients en attente+ nombre de clients étant servis
- Critère d'arrêt de simulation: temps de simulation maximal = 60 mins

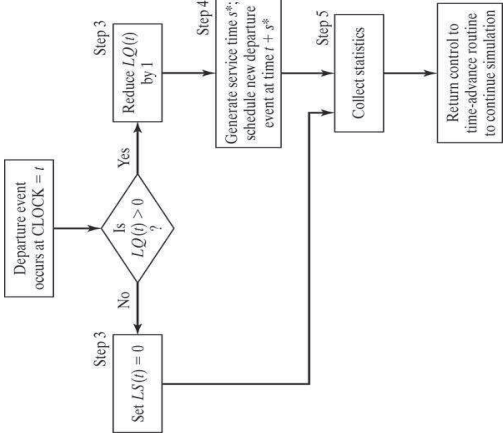


## Exemple d'épicerie

Composants du modèle:

- Notifications d'événements (type d'événement, heure de l'événement):
  - (A, t) représentant un événement d'arrivée à se produire à l'instant futur t.
  - (D, t) représentant un départ du client à l'avenir t.
  - (E, 60) représentant l'événement d'arrêt de simulation à l'avenir 60.
- Activités: heure d'arrivée et temps de service.
- Retard: temps passé par le client dans le fils d'attente.





### Exemple d'épicerie

Les conditions initiales sont « le premier client arrive à t=0 » et « commence le service ».

- Deux statistiques seulement:
- l'utilisation du serveur (B)
  - la longueur maximale de la file d'attente (MQ).

### Exemple d'épicerie

Horloge t	LQ(t)	LS(t)	LEF	B	MQ
0	0	1	(D,4)(A,8) (E,60)	0	0
4	0	0	(A,8) (E,60)	4	0
8	0	1	(D,9)(A,14)(E,60)	4	0
9	0	0	(A,14)(E,60)	4	0
14	0	1	(A,15)(D,18)(E,60)	5	0

### Exemple d'épicerie

Horloge t	LQ(t)	LS(t)	LEF	B	MQ
15	1	1	(D,18)(A,23) (E,60)	6	1
18	0	1	(D,21)(A,23)(E,60)	9	1

### Pratique de simulation

### Pratique de simulation

- Assurer de bonnes pratiques lorsque la modélisation de simulation est importante pour obtenir des analyses significatives à partir des modèles.

## Pratique de simulation

- Considérons une banque ouverte 24/7 et on la simule pour une semaine.
- Cela signifie que la banque ne partirait jamais d'un état vide, comme le fait notre simulation.
- Ces clients sont partis à l'intérieur du système à la fin de la course : leurs enregistrements n'ont pas été collectés, malgré le temps passé dans le système pendant la période d'observation.
- Est-ce que cette simulation unique reflète vraiment la réalité? Nos résultats étaient simplement un coup de chance? Un cas extrême?

## Effectuer des répétitions multiples

- Les utilisateurs ne doivent pas s'appuyer sur les résultats d'une seule exécution de la simulation en raison de la nature stochastique intrinsèque de la simulation.
- Lorsque vous exécutez une seule répétition, les utilisateurs ne peuvent pas savoir si le comportement de cette exécution est typique, extrême ou inhabituel.
- Pour contrer ces multiples répétitions doivent être effectuées, chacune utilisant des flux de nombres aléatoires différents.
- Ensuite, des analyses sur la distribution des résultats peuvent être effectuées (par exemple en prenant des valeurs moyennes d'indicateurs clés de performance).

## Effectuer des répétitions multiples

- Dans Ciw, la façon la plus simple d'implémenter ceci est de créer et d'exécuter la simulation en boucle, en utilisant une graine aléatoire différente à chaque fois.

ciw.seed(s)

## Temps de préchauffage

- Les modèles de simulation commencent souvent dans des circonstances irréalistes, c'est-à-dire qu'ils ont des conditions initiales irréalistes.
- Dans Ciw, la condition initiale par défaut est un système vide.
- Bien sûr, il peut y avoir des situations où la collecte de tous les résultats d'un système vide est requise, mais dans d'autres situations, par exemple lors de l'analyse des systèmes en équilibre, ces conditions initiales provoquent un biais indésirable.

## Temps de préchauffage

- Une méthode standard pour surmonter cela est d'utiliser un temps de préchauffage.
- La simulation est exécutée pendant un certain temps (le temps de préchauffage) pour obtenir le système dans un état approprié avant que les résultats soient collectés.
- Dans Ciw, la façon la plus simple de mettre en œuvre ceci est de filtrer les enregistrements qui ont été créés pendant la période de préchauffage.

## Temps de recharge

- Si vous collectez des enregistrements à l'aide de la méthode *get\_all\_records*, seuls les enregistrements terminés seront collectés.
- Il peut être nécessaire de recueillir des informations sur l'arrivée ou l'attente de ces personnes encore en service.

## Temps de recharge

- Dans Ciw, nous pouvons le faire en simulant au-delà de la fin de la période d'observation, puis seulement collecter les enregistrements pertinents qui sont dans la période d'observation.
- Dans Ciw, la façon la plus simple de mettre en œuvre ceci est de filtrer les enregistrements qui ont été créés après le début du refroidissement.

## Exemple : file d'attente M/M/1

```
import ciw
N = ciw.create_network(
    Arrival_distributions=[['Exponential', 5.0]],
    Service_distributions=[['Exponential', 8.0]],
    Transition_matrices=[[0.0]],
    Number_of_servers=1
)
average_waits = []
warmup = 10
cooldown = 10
maxsimtime = 40
```

## Exemple : file d'attente M/M/1

```
for s in range(25):
    ciw.seed(s)
    Q = ciw.Simulation(N)
    Q.simulate_until_max_time(warmup + maxsimtime + cooldown)
    recs = Q.get_all_records()
    waits = [r.waiting_time for r in recs if r.arrival_date > warmup
              and r.arrival_date < warmup + maxsimtime]
    average_waits.append(sum(waits) / len(waits))
```

## Exemple : file d'attente M/M/1

```
average_wait = sum(average_waits) / len(average_waits)
average_wait
```