Évaluation de Performances

Simulation

Florence Perronnin

Université Joseph Fourier

April 12, 2016

Outline

Les chercheurs, les ingénieurs, et bien d'autres professionnels se posent souvent la question : quel est le résultat que j'obtiens si j'exerce telle action sur un élément ?

Tenter l'expérience :

- irréalisable
- trop cher
- contraire à l'éthique

Recours à la simulation :

- système qui réagit d'une manière semblable
- permet de déduire des résultats.

Definition

Une simulation est une imitation d' un système réel. Elle repose sur un modèle de la réalité qui sélectionne les caractéristiques clés et les fait évoluer selon la dynamique du modèle.

Applications

La simulation est utilisée dans beaucoup de contextes:

- modélisation de systèmes naturels ou humains
- optimisation des performances de technologies
- tests de sécurité
- systèmes d'entraînements

Example

- simuler une opération chirurgicale
- évaluer une nouvelle politique commerciale
- résistance d'une plateforme pétrolière à la houle
- calculer un risque sismique
- quantifier les performances d'une politique d'ordonnancement

Simulation par ordinateur

Definition

Une simulation numérique (dite aussi simulation informatique) est une expérience dans laquelle l'objet étudié et son environnement sont remplacés par l'exécution d'un programme informatique.

Les simulations numériques scientifiques reposent sur la mise en œuvre de modèles théoriques discrets. Elles sont donc une **adaptation aux moyens numériques de la modélisation mathématique**.

Definition

Une simulation est une expérience visant à prédire le comportement d'un système complexe (ex: chute d'un corps sur un support mou, évolution de la météo, comportements de systèmes biologiques) en implémentant un modèle approximatif de celui-ci.

Le problème de la validité

La simulation numérique tend à devenir un outil indispensable dans le domaine industriel d'ingénierie des objets complexes du fait de sa rapidité de mise en œuvre (ex: simulation de crashes automobiles, exploration des domaines de vol aéronautiques, simulation d'avalanches, ...).

Elle ne doit pas être utilisée sans prise de recul scientifique, ni prudence professionnelle.
L'analyse critique des résultats, la vérification de la validité des modèles théoriques utilisés, la confrontation des résultats prédits à l'expérience ... sont autant de réflexes que le chercheur doit avoir et qui font partie de l'éthique du professionnel utilisateur.

Avant de simuler...

Avantages

- reproductible sur différentes configurations
- estimer des quantités difficiles à mesurer in situ
- accélération (temps simulé)
- gain d'argent (équipement)
- n'interfère pas avec le système opérationnel
- Peut répondre à "what if?"
- plus réaliste qu'un modèle analytique

Inconvénients

- bugs
- effet des conditions initiales
- système non stationnaire
- mauvais critères d'arrêt
- coûteux à développer
- mauvaise analyse statistique des résultats
- mauvais modèle

Validation

Definition

Valider signifie s'assurer que le modèle simulé est représentatif de la réalité et donne des résultats fiables.

- valider les hypothèses
- représentativité des input
- validation du modèle
- ergodicité
- réalisme et explicabilité des résultats

Méthodes de validation

- comparaison avec un système réel sur un benchmark
- modèle analytique
- raisonnement

Vérification

Mais aussi...

Definition

Vérifier signifie s'assurer que le modèle choisi a été correctement implémenté (structures, bugs).

- validation de logiciel
- validation stochastique du générateur aléatoire

Définitions

Definition

le temps simulé est l'instant auquel un événement du système réel a lieu.

Les différents rôles du temps simulé :

- contrôle : Le temps simulé permet de contrôler le flot d'exécution. (Ex: action simultanées traitées séquentiellement)
- durée : La durée effective d'une simulation est sans commune mesure avec la durée simulée. (Ex: évènements consécutifs séparés d'un temps très grand ou infinitésimal)
- manipulation du temps : le temps dirige la simulation:
  - simulation équationnelle (récurrente): calcul à chaque instant d'arrivée
  - simulation à événements discrets : échéancier
- propriétés : le temps peut être discret ou continu

Définitions

Definition

un événement est un changement d'état du système (aussi appelé transition).

Les événements sont les instants intéressants d'une simulation. Le nombre d'événements va influencer le temps de calcul de la simulation. On parle de simulation à événements discrets lorsque l'espace d'états est discret et que la simulation est basée sur les événements.

Example

arrivées et départs de paquets dans une M/GI/1

Contre-exemple

simulation de l'écoulement d'un fluide.

Stationnarité

Definition

un processus $X(t)$ est stationnaire si et seulement si la distribution conjointe de $X(t_1), X(t_2), ..., X(t_n)$ est identique à la distribution conjointe de $X(t_1 + h), X(t_2 + h), ..., X(t_n + h)$.

Propriétés

- moyenne invariante : $\mathbb{E}[X(t)] = \mu$
- variance invariante : $V[X(t)] = \sigma^2$
- autocorrélation invariante : $\rho(t, \tau) = \rho(\tau)$

Attention à la dépendance aux conditions initiales et à la durée de simulation! (stationnarité asymptotique)

Critère d'arrêt

évènement terminal

Évènement dont l'occurrence met fin à la simulation. Son observation constitue l'objectif de la simulation.

Example

durée de transfert d'un fichier, état d'une pièce après 10 000 torsions, etc.

Lorsqu'il n'existe pas d'évènement terminal, il faut déterminer à quel instant (simulé) l'on doit arrêter la simulation.

critère d'arrêt

Un définit les conditions dans lesquelles le système simulé se trouve pour terminer le programme de simulation.

État stationnaire

Problème

Lorsqu'une simulation n'a pas d'événement terminal, il faut s'assurer de la stationnarité du système observé, afin de ne pas dépendre de l'âge et des conditions initiales.

Méthodes possibles:

- Points de régénération
- Simulation parfaite — Propp et Wilson
- Tests de stationnarité
- Élimination de la période transiente

Points de régénération

Definition

un processus stochastique $X(t)$ est un processus de régéneration (regenerative process) s'il est composé de cycles i.i.d. [Asmussen]

les états séparant les cycles sont appelés des points de régéneration.

Example

Pour une file d'attente M/M/1, les instants où la file est vide sont des points de régénération.

On peut alors simuler le système sur un cycle complet pour s'affranchir des conditions initiales.

Processus non stationnaires

- recherche de tendances :
  - somme d'une fonction déterministe du temps (tendance) et d'un processus aléatoire stationnaire
- décomposition en intervalles de temps sur lesquels le processus est stationnaire (2 échelles de temps)

Outline

Types de modèles

Rappel: une simulation est un programme qui déroule le comportement d'un modèle du système étudié. Selon les cas, ce modèle peut être:

- aléatoire vs déterministe
  - Arrivées de clients / propagation d'une onde de choc
- avec ou sans événement terminal
  - durée d'exécution vs temps moyen de réponse
- asymptotiquement stationnaire / non stationnaire
- statique/dynamique
  - Monte-Carlo vs évolution d'un système

Types de Simulation

1. simulation à événements discrets (event-driven)
2. simulation de traces (trace-driven).

Avantages:

- crédible
- validation
- charge réaliste
- résultats détaillés
- comparaison équitable

Inconvénients:

- complexité
- représentativité de la trace
- finitude (ex: 1 journée)
- pas de variation possible
- biais système/trace

3. simulation équationnelle (stochastic recurrence)
4. simulation Monte Carlo

Outils de simulation

Choix entre 3 techniques:

- programme ad hoc dans un langage générique :
  - C, C++, Java...
- programme ad hoc dans un langage de simulation :
  - SIMULA, CSIM, SIMSCRIPT GPSS...
- simulateur existant :
  - simulateurs de réseaux : ns2, NSE,daSSF...
  - simulateurs d'applications : LAPSE, SimGrid...

Choix de l'outil

Attention à vérifier l'adéquation de l'outil aux besoins!

- temps de formation
- temps de développement
- finesse du simulateur...

Outline

Principe de la simulation Monte-Carlo

On appelle méthodes de Monte-Carlo les procédés visant à calculer une valeur numérique en utilisant des quantités aléatoires.

Le véritable développement des méthodes de Monte-Carlo s'est effectué, sous l'impulsion de Stanislaw Ulam, Enrico Fermi et John von Neumann lors de la Seconde Guerre mondiale et des recherches sur la fabrication de la bombe atomique. Notamment, ils ont utilisé ces méthodes probabilistes pour résoudre des équations aux dérivées partielles.

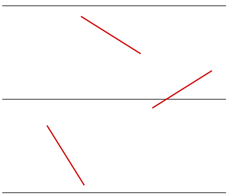
Les aiguilles de Buffon

Calcul de π

Comment calculer π par une méthode de Monte-Carlo?

Algorithme de Buffon

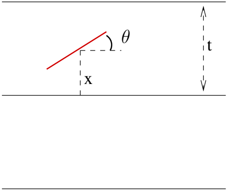
On fait tomber plusieurs aiguilles de longueur ℓ sur un plancher dont les lames sont de largeur t (avec $t > \ell$) et on calcule la proportion d'aiguilles qui reposent sur plusieurs lames.



Soit x la distance de l'aiguille à la plus proche jointure et θ l'angle aigu entre l'aiguille et les lames. L'aiguille touche deux lames dès que

$$x \leq \frac{\ell}{2} \sin \theta.$$

Les aiguilles de Buffon(II)



Remarquons que x est uniformément distribué sur $[0, t/2]$, θ est uniformément distribué sur $[0, \pi/2]$ et ils sont indépendants. La probabilité que l'aiguille touche deux lames vaut

$$\int_0^{\frac{\pi}{2}} \int_0^{(\ell/2)\sin\theta} \frac{4}{t\pi}\, dx\, d\theta = \frac{2\ell}{t\pi}.$$

En faisant tomber N aiguilles et en notant h_N le nombre d'aiguilles qui traversent une jointure, on obtient

$$\pi = \lim_{N\to\infty} \frac{N}{h_N}\frac{2\ell}{t},$$

en utilisant la loi des grands nombres.

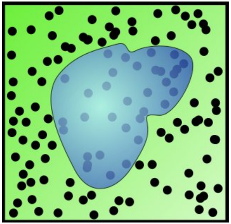
L'expérience de Lazzarini

Mario Lazzarini un mathématicien Italien a réalisé l'expérience de Buffon en 1901. En lançant 3408 aiguilles, il a obtenu comme estimation de π, $355/113$, soit une erreur inférieure à 2.10^{-7} !

En fait, on peut penser que Lazzarini a biaisé son expérience...
Il avait choisit un rapport $\ell/t = 5/6$. Dans ce cas, on obtient $\pi \approx 5/3 \times N/h$.
Si de plus, on fait en sorte que $h = 113N/213$, alors on obtient $355/113$ comme approximation de π, qui s'avère être la meilleure approximation rationnelle de π avec des facteurs de moins de 5 chiffres.

Or $3408 = 213 \times 16$...

Calculs d'intégrales



En général, la précision du calcul est en $O(1/\sqrt{N})$, par le Théorème Central Limite. Cette technique (amélioré par l'utilisation de lois non-uniformes) a été utilisée à Los Alamos pour faire les calculs nécessaires à la mise au point de la fusion nucléaire.

Remarque

Ces techniques ont besoin d'un générateur aléatoire de grande qualité.

Échantillonner une loi

Le principe: calculer des échantillons d'une loi de probabilité.
On considère une variable aléatoire X finie, qui suit une loi p.
($X = x_i$ avec probabilité p_i, pour $i = 1, \cdots, N$).
Pour simuler X, on peut utiliser la technique suivante:

On génère une variable aléatoire U uniforme sur $[0, 1]$ et on calcule $X = \phi(U)$ avec

$$\phi(u) = \begin{cases} x_1 & \text{si } u \in [0, p_1[, \\ x_2 & \text{si } u \in [p_1, p_1 + p_2[, \\ \vdots & \vdots \\ x_N & \text{si } u \in [p_1 + p_2 + \cdots + p_{N-1}, 1]. \end{cases} \tag{1}$$

Dans la pratique cette technique est inutilisable quand la taille de l'espace d'état, N, est grande.

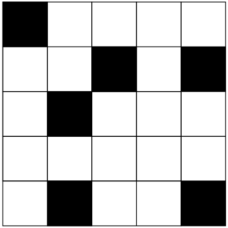
Monte Carlo Markov Chain: principe

L'idée de la simulation MCMC est de construire une suite de variables aléatoires $X_0, X_1 \dots X_n \dots$ dont la loi limite est p, indépendamment de X_0. Puis de simuler cette suite (chaîne) pendant un temps assez long.

Il est légitime de se poser la question: comment cela peut-il être plus efficace que de construire une variable aléatoire dont la distribution est directement p?

Simulation Monte-Carlo sur un exemple

Soit un quadrillage d'un carré $N \times N$ du plan. Une configuration du système donne les valeurs 0 ou 1 à toutes les cases du carré, sans que deux 1 ne soient adjacents.



On veut calculer une configuration qui suit la loi uniforme: Si on note Z le nombre de configurations possibles, alors chacune a une probabilité $1/Z$.

Ce problème est difficile à résoudre de manière directe (déjà le calcul de Z n'est pas facile...).

Simulation Monte-Carlo: échantillon de Gibbs

Voici une solution qui ne nécessite pas le calcul de Z.
On choisit X_0 (une configuration initiale valide quelconque, par exemple, toutes les cases sont à 0). Et on calcule X_{i+1} de la manière suivante.

- On choisit une case au hasard.
- On tire à pile ou face.
- Si pile alors la case passe à 1 (si c'est autorisé).
- Si face, la case passe à 0.

Théorème

Si on itère un grand nombre de fois, on obtient une configuration typique du système.

Outline

Simulation à événements discrets

La simulation à événements discrets simule des systèmes à états discrets mais qui peuvent être à temps continu ou à temps discret.

Example

- Une file GI/GI/1 est un système à états discrets et à temps continu.
- Un écoulement de fluide n'est pas à états discrets.
- Une marche aléatoire (ex: TD Tom & Jerry) est à états discrets et à temps discret.

Architecture

Composants classiques :

- État du système (variables)
- Routines pour chaque événement
- Initialisation
- Récupération des input
- Enregistrement (traces, stats)
- Gestion de la mémoire
- Programme principal...

Composants SED

- horloge (variable temps)
- mécanisme d'avancement du temps
- échéancier (scheduler)
- noyau de synchronisation
- bootstrap
- critère d'arrêt

Programme principal

Bootstrap {Crée le 1^{er} événement et l'insère dans l'échéancier.}
repeat
avancer l'horloge à l'instant t du prochain événement e
Mettre à jour les variables dépendant du temps t
Exécuter e {actions sur l'état et insertion/suppression d'événements dans l'échéancer}
Retirer e de l'échéancier
until critère d'arrêt satisfait
Terminaison {calculer les statistique finales et produire le rapport}

Noyau de synchronisation

Le noyau de synchronisation est le module qui effectue les opérations sur l'échéancier :

- insertion d'événement
- accès au prochain événement
- suppression du prochain événement
- annulation d'un événement quelconque
- maintenir événements non datés
- dater un événement
- classer les événements simultanés

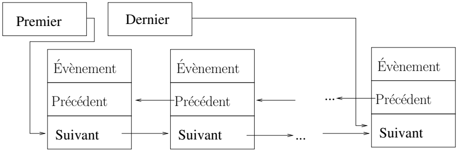
Ce module est le plus féquemment exécuté dans le programme. Son optimisation est directement liée à la structure de l'échéancier.

Échéancier

L'échéancier est la structure de données qui stocke les événements potentiels.

Le choix de sa structure dépend des caractéristiques du modèle:

- liste (doublement) chaînée : coût de recherche
- liste chainée indexée par intervalle de tps (ex calendrier annuel)
- tableau trié : coût d'insertion
- arbre : si beaucoup d'événements



Efficacité

L'efficacité du noyau de synchronisation tient à:

- l'adéquation de la structure de l'échéancier au modèle simulé
- l'agorithme d'entretien de l'échéancier
- le nombre d'événement stockés simultanément
- la distribution du temps entre les événements

Outline

Systèmes ergodiques

Dans la plupart des systèmes considérés, (en particulier en mécanique statistique) les quantités d'intérêt concernent les moyennes définies sur l'ensemble des états accessibles (ou configurations). Une simulation Monte-Carlo ou de dynamique moléculaire donne une moyenne temporelle, sur une trajectoire d'un système dynamique.

Ces deux moyennes ne coïncident, que si le système est *ergodique* (Von Neumann, Gibbs, Birkhoff (1931)).

Ainsi sans la condition d'ergodicité, les simulations numériques deviennent caduques.

Croissance des villes et la loi de Zipf

Un système sur n sites suit une loi de Zipf si la taille d'un site est inversement proportionelle à son rang. Elle a été observée dans la fréquence d'apparition des lettres dans la langue Anglaise, mais aussi à l'échelle nationale et mondiale sur les population des villes.

En 1955, Herbert Simon a conçu un modèle de croissance qui repose sur trois hypothèses simples et qui semble adapté pour décrire des systèmes naturels et sociaux, en particulier celui de la croissance des villes.

Les n sites (ou villes) ont une population $T_1 \geq T_2 \geq \cdots \geq T_n$. Lors de l'arrivée d'un nouvel habitant, il fonde un nouveau site avec probabilité α ou rejoint la ville i avec probabilité proportionnelle à $(1-\alpha)T_i$.

Un tel système suit une loi de Zipf limite, avec un paramètre variable et dépendant de α.

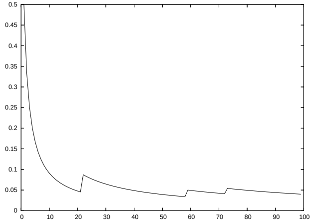
Simulation de la croissance des villes

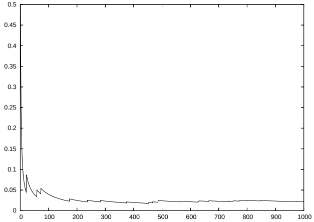
On simule le modèle de Herbert Simon avec deux villes et $\alpha = 0$ (aucune nouvelle ville n'est créée):
On génère une variable aléatoire U, uniformément sur $[0, 1]$.
Si $U < T_2/(T_1 + T_2)$ alors, le nouvel habitant s'installe dans la ville 2.

Si la taille initiale des villes vaut $T_1(0) = T_2(0) = 1$, que se passe-t-il après un grand nombre d'arrivées? On s'intéresse à la proportion asymptotique d'habitants dans la plus petite ville: $\lim_{n\to\infty} T_2(n)/(T_1(n) + T_2(n))$.

Simulation de la croissance des villes (II)

On simule le système sur 100 arrivées, puis pour améliorer la précision, sur 900 arrivées supplémentaires.



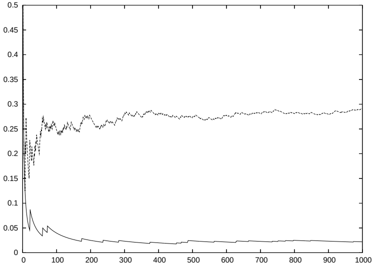


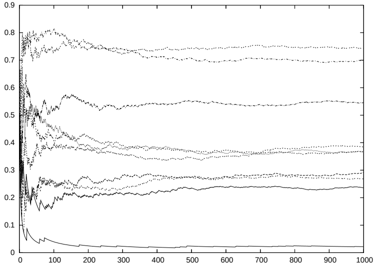
Conclusion

Après un temps assez long, la proportion de la population dans la deuxième ville est de l'ordre de 3%.

Simulation de la croissance des villes (III)

On refait la simulation plusieurs fois:





Explication

Cette variable n'est pas ergodique, la moyenne de Cesaro ne converge pas vers l'espérance.

Théorème

Outline

Conclusion

Erreurs classiques

Common mistakes in simulation [Jain].

- attention aux bugs
- valider la simulation
- adapter le type de simulation au système étudié
- connaître d'avance les types d'output à donner
- vérifier la stationarité du processus
- représentativité des entrées
- choix des conditions initiales
- coût du projet
- optimiser les E/S (enregistrement de traces notamment)

Sources

Bibliographie

- Asmussen and Glynn, *Stochastic Simulation. Algorithms and Analysis,* Springer-Verlag, 2007.
- Banks, Carson, Nelson and Nicol, *Discrete-Event System Simulation*, Prentice Hall, 2001.
- Jain, *The Art of Computer Systems Performance Analysis*, Wiley, 1991.
- Le Boudec, *Performance Evaluation of Computer and Communication Systems*, 2006.
- B. Gaujal: *La Simulation en Science,* 2006.