Performance Modeling of Computer Systems and Networks

Prof. Vittoria de Nitto Personè

Simulation introduction

Università degli studi di Roma Tor Vergata
Department of Civil Engineering and Computer Science Engineering

1

Performance evaluation techniques

Computational and mathematical techniques to *model*, *simulate* and *analyze* the performance of $\underbrace{\textit{stochastic}}_{\text{random/aleatorio}}$ *systems*

Modeling: conceptual framework describing a system (struttura)

Simulate: perform experiments using computer implementation of the model

Analyze: draw conclusions from output

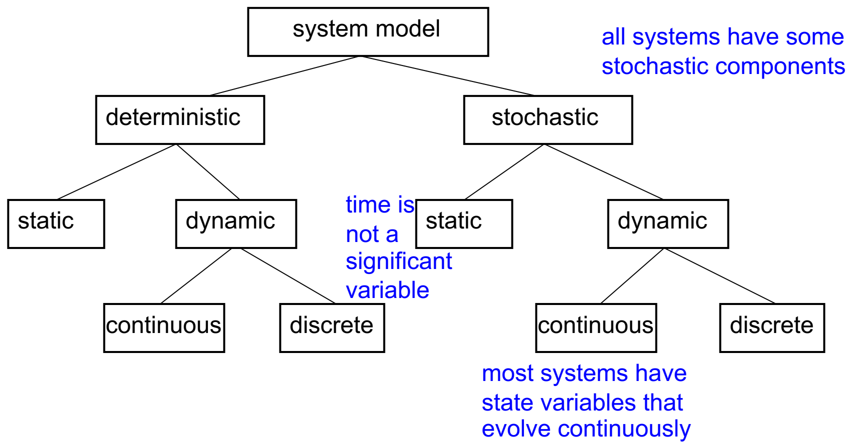
Simulation models

Analytical models

2

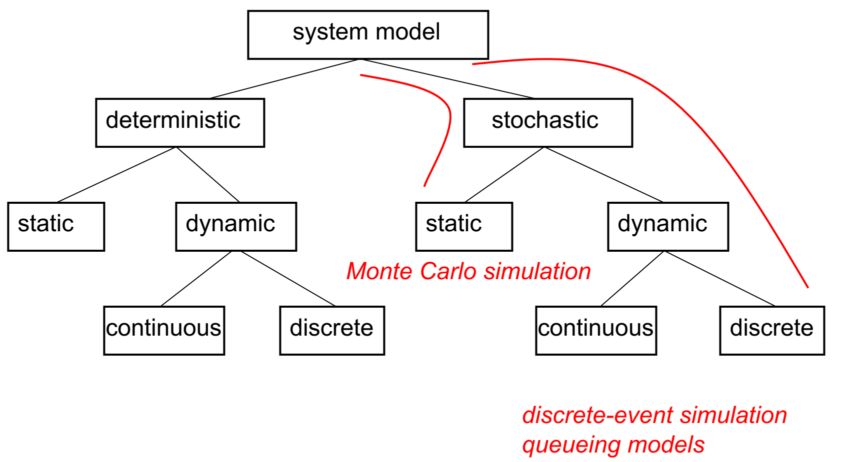
P.13 discrete

Characterizing a model



3

Characterizing a model



4

model development

Algorithm 1.1: how to develop a model

(es: serve 5° server? quanti server occorrono?)

1. Goals and objectives e.g. Boolean decisions Numeric decisions **UNBIASED**
2. *Conceptual* model (cm) → var. stato? sono legate? quali sono importanti?)
3. Convert cm into a *specification* model (sm) collezione/analisi per fornire input models utili per la simulazione.
4. Convert sm into a *computational* model (cptm) gen. purpose OR special purpose?
5. Verify (implementazione del cptm corretta?)
6. Validate (modello è corretto?)

Prof. Vittoria de Nitto Personè 5

5

model development

Three Model Levels

i. Conceptual
   - very high level
   - which are the state variables, how they are related, which can be ignored and which not

ii. Specification
   - On paper
   - May involve equations, pseudocode, etc.
   - How will the model receive input?
     - collecting and statistically analyzing data
     - using representative stochastic models

iii. Computational
   - A computer program
   - General-purpose PL or simulation language?

Prof. Vittoria de Nitto Personè 6

6

Verification vs. Validation

5. Verification
   - Computational model should be consistent with specification model
   - Did we build the model right?
6. Validation
   - Computational model should be consistent with the system being analyzed
   - Did we build the right model?
   - Can an expert distinguish simulation output from system output?

Algorithm 1.1: how to develop a model

1. Goals and objectives e.g. Boolean decisions
   Numeric decisions **UNBIASED**
2. *Conceptual* model (cm)
3. Convert cm into a *specification* model (sm)
4. Convert sm into a *computational* model (cptm)
5. Verify
6. Validate

Typically an iterative process

model development

Algorithm 1.1: observations

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate

- Make each model as simple as possible:
  - Never simpler
  - Do not ignore relevant characteristics
  - Do not include extraneous characteristics
- Model development is not sequential
  - Steps are often iterated
  - For teams, steps may be in parallel
  - Do not merge verification and validation
- Develop models at three levels
  - Think a little, program a lot (and poorly);
  - Think a lot, program a little (and well).

~~1. Goals~~
~~2. Conceptual model~~
~~3. Specification model~~
~~4.~~ 1. Computational model
~~5.~~ 2. Verify
~~6.~~ 3. Validate

Certainly produce large, inefficient, unstructured cm that CANNOT BE VALIDATED

Prof. Vittoria de Nitto Personè 9

9

Simulation studies

Algorithm 1.2: using the resulting model

7. Design simulations experiments
   - What parameters should be varied?
   - perhaps many combinatoric possibilities
8. Make production runs
   - Record initial conditions, input parameters
   - Record statistical output
9. Analyze the output
   - Random components → statistical analysis (means, standard deviations, percentiles, histograms etc.)
10. Make decisions
    - The step9 results drive the decisions → actions
    - Simulation should be able to correctly predict the outcome of these actions (→ further refinements)
11. Document the results
    - summarize the gained insights in specific observations and conjectures useful for subsequent similar system models

Prof. Vittoria de Nitto Personè 10

10

Machine Shop Model

- 150 identical machines:
  - Operate continuously, 8 hr/day, 250 days/yr
  - Operate independently
  - Repaired in the order of failure
  - Income: 50,00 €/hr of operation
- Service technicians:
  - 2-year contract at 60.000,00 €/yr
  - Each works 230 8-hr days/yr

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate

- How many service technicians should be hired to maximize the profit?

Extreme solutions: just 1 technician
→ minimizes service-techn overhead
→ large down-times
→ loss of income

1 technician for each machine
→ huge service-techn overhead
→ minimum down-times
→ maximizes income

model development
example

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate

Job
SS

- State of each machine (failed, operational)
- State of each techn (busy, idle) ≈ multi-server
- Provides a high-level description of the system at any time

Prof. Vittoria de Nitto Personè 13

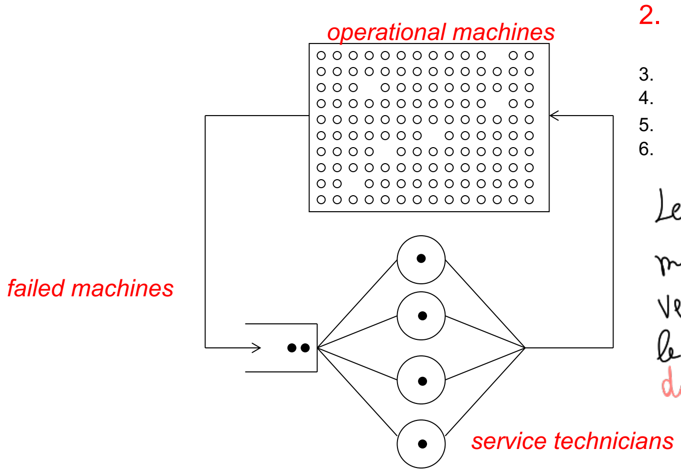
13

P.16)

model development
example

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate



Le prime 4 macchine guaste vengono riparate, le altre 2 (rotte dopo) attendono

Prof. Vittoria de Nitto Personè 14

14

Machine Shop Model

1. Goals
2. Conceptual model
3. **Specification model** (collezione DATI)
4. Computational model
5. Verify
6. Validate

- What is known about time between failures? Are the failures random?
- What is the distribution of the repair times?
- How will time evolution be simulated?

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. **Computational model**
5. Verify
6. Validate

- It should include:
  - Simulation clock data structure
  - «Queue» of failed machines
  - «Queue» of available technicians
  - performance characterization (structures to collect statistical data)

model development
example

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate

- Software engineering activity
- Usually done via extensive testing

] ING. software

Prof. Vittoria de Nitto Personè 17

17

model development
example

Machine Shop Model

1. Goals
2. Conceptual model
3. Specification model
4. Computational model
5. Verify
6. Validate

the validation step allows to verify if the cptm is a "good approximation" of the actual machine shop

- If operational, compare against the real thing

- otherwise → use *consistency checks*

e.g. as the n. of technies grows, the average n. of fault machines decreases

as the mean service time grows, the average n. of fault machines grows too

es: • se n° tecnici ↗, n° macchine guaste ↙

• se tempo riparazione ↗, n° macchine guaste ↗

Prof. Vittoria de Nitto Personè 18

18

Machine Shop Model

7. Experiments design
8. Runs production
9. Output analysis
10. Decisional phase
11. Results documentation

- Find the optimal number of technies to maximize profit
- Initial conditions (e.g. are all machines initially operational?)
- For a fixed n. of service technies, how many replications are required to reduce the natural sampling variability in the output statistics to an acceptable level?

variabilità del campionamento (posso avere RUN influenzati da scenari diversi, prob. guasto...)

Machine Shop Model

7. Experiments design
8. Runs production
9. Output analysis
10. Decisional phase
11. Results documentation

- If many runs are made,
  management of the output results becomes an issue
  → avoid to archive "raw date" (dati grezzi)
  simulation advantage: experiments can always be reproduced

Machine Shop Model

7. Experiments design
8. Runs production
9. **Output analysis**
10. Decisional phase
11. Results documentation

- The statistical analysis (sa) of sim output often is more difficult than classical sa
  → *dependent (/correlated)* observations
  e.g. if the current n. of failed machines is observed each hour, consecutive observations will be found positively correlated → both below or above the mean n. of failed machines
- ATTENTION to erroneous conclusions

Il campione deve caratterizzare la popolazione, esso è creato secondo metodi IID (Indipendenti & Identicamenti Distribuiti), ben costruito!

Machine Shop Model

7. Experiments design
8. Runs production
9. Output analysis
10. **Decisional phase**
11. Results documentation

- A graphical display of profit versus the number of service technies yields both the optimal n. of technies and a measure of how sensitive the profit is to variations of this n. (cost)
- *Decision policy not violating any external constraint*

Simulation studies
example

Machine Shop Model

7. Experiments design
8. Runs production
9. Output analysis
10. Decisional phase
11. Results documentation

- System diagram
- Assumptions about failure and repair rates
- Description of specification model
- Software
- Tables and figures of output
- Description of output analysis

Advantages of the sim study:
can provide valuable insights about system features and component interactions otherwise not achievable

terminology

- *Model / simulation* (noun)
  - Model can be used with respect to conceptual, specification, or computational levels and for both analytical and simulation techniques
  - Simulation is frequently used to refer to the computational model (program), it is rarely used to describe the conceptual or specification model
- *Model / simulate* (verb)
  - To model can refer to development of the levels
  - To simulate refers to the computational activity
- ATTENTION do not confuse *verify* with *validate*

Exercises

- Ex 1.1.2 and Ex 1.1.3 on p.11 from textbook

1.1.2)

25

The distinction between model verification and model validation is not always clear in practice. Generally, in the sense of Algorithm 1.1.1, the ultimate objective is a valid discrete-event simulation model. If you were told that "this discrete-event simulation model had been verified but it is not known if the model is valid" how would you interpret that statement?

- La simulazione si comporta come previsto (es: somma 2+2=4) ma non c'è sicurezza sul fatto che il modello sia quello voluto! (es: $\underbrace{\text{somme OK}}_{\text{verify}}$, ma io volevo fare $\underbrace{\text{moltiplicazioni}}_{\text{validate}}$!)

1.1.3)

The state of a system is important, but difficult to define in a general context. (a) Locate at least five contemporary textbooks that discuss system modeling and, for each, research and comment on the extent to which the technical term "state" is defined. If possible, avoid example-based definitions or definitions based on a specific system. (b) How would you define the state of a system?

- ★ caratterizzazione system ad un istante t, con valori delle VAR, definisce funzioni eseguibili in t (ISO)
- valore assegnato ad un attributo è costante/stabile in un Δt (INCOSE)
- informazione riguardante una cosa (oggetto), in un tempo t, in un contesto temporale (Naumenko)
- Attributo caratterizzante la condizione del sistema, basata su performance e condizioni del sistema.

} badvel_24780.pdf by oatao.univ