S-38.3148 Simulation of data networks

Theory

- Introduction to simulation
  - Simulation as a method for performance analysis
  - Classification of simulation models
  - Modeling, implementation and validation
- Flow of simulation -- generating process realizations
- Random number generation from given distribution
- Collection and analysis of simulation data
- Variance reduction techniques

What is simulation?

- With the aid of a computer program one imitates or *simulates* the operation of some real equipment or process
- The equipment or process is called the *system*
- In order to study the system, one has to make some assumptions about how the system operates
- The assumptions constitute the *model* of the system
  - typically the assumptions take the form of mathematical or logical relations

What is simulation (continued)?

- If the mathematical relations contained in the model are simple it may be possible to obtain *exact* information about the system by a mathematical analysis
- Most systems of the real world, however, are far too complicated for an exact mathematical analysis
- In such cases one may use simulation
  - in some sense simulation is “the last resort”
  - often it is only possible alternative
- In simulation a computer program is used for *numerical* imitation of the system
- In the course of simulation data is collected in order to *estimate* the interesting quantities

Objectives of the simulation of telecommunication systems

- Research and development
  - Improvement of the performance
    - delays, throughput etc
  - Identifying the system bottlenecks before actual implementation of the system
  - Shortening the development time of the system
- Planning operations
  - Guaranteeing performance goals before making purchasing decisions
  - Estimating the effects of “local” changes on the whole network
- In general: “minimizing the costs”
  - testing on real equipment/systems is typically more expensive than simulation

Typical questions to be answered by simulation

- Traffic performance:
  - How does the network perform under increased traffic?
  - How big buffers are needed in the routers?
  - Is the capacity of the web server big enough to sustain an increased load without the response times increasing too much?
  - Which TCP ”variant” gives the best performance?
- Network topology:
  - What is the best network topology?
  - Should a company’s network be partitioned into subnetworks due to increased number of workstations?
  - How many satellites are needed to achieve a given quality of service between two ground stations?
- Network reliability:
  - What happens when a link fails in the network?
  - How quickly does routing converge after link failures?

Typical quantities of interest in simulations

- Throughput (Mbit/s)
- End to end delay
- Delay between two points, A and B, in the network
- Number of packets in a buffer
- The utilization of links
- The blocking probability of a connection
- The probability of failure of a call hand-over in a mobile network
- The collision probability of packets in a LAN

Advantages of simulation

- One can study the operation of very complicated systems
- One can study the operation of a system in the planning stage (without building a real system or a prototype)
  - compare different planning options or operation policies
  - sometimes building the simulation model is more important than the simulation itself
- One can study the behavior of an existing system under different operation conditions
  - by means of simulation one can tune the system parameters more easily and accurately than in the real system
- Simulation allows one to study the behavior of the system over longer periods of time (in a compressed time scale)
  - or conversely very fast events in a slowed down time scale

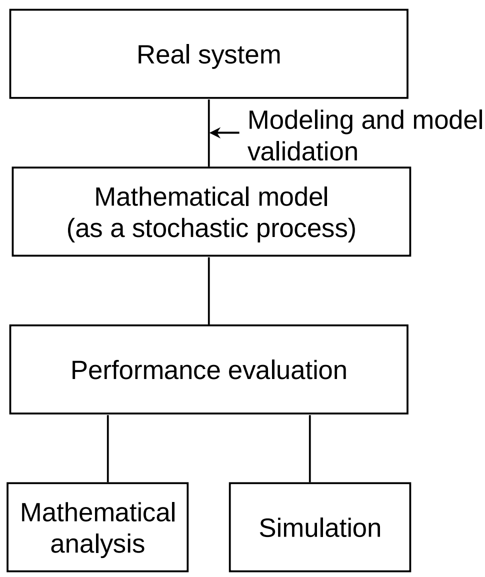
... and limitations

- In case of very large systems, even simulation approach may not be feasible because
  - the creation of the simulation model may require too much effort
  - writing the simulation program may be a very big task
    - several simulation languages and tools are available to facilitate the task
  - running the simulation program may take a very long time

- Simulation does not solely or primarily concern writing the program
  - important is the model and
  - how simulation is used to make conclusions

Performance evaluation: analysis vs. simulation

- ***Simulation*** is (from the point of view of traffic theory) a statistical method for the performance evaluation of the considered system
- It consists of four steps
  - the modeling of the system (a real one or a virtual one) as a dynamic stochastic process (developing in time)
  - generation of realizations of the process (“observation“)
  - collection of data (“measurement”)
  - statistical analysis of the data and drawing conclusions
- ***Mathematical analysis*** in an alternative approach
- It consists of two steps
  - the modeling of the system as a time-dependent stochastic process
  - analytical solution of the model
- Modeling the system is common for both approaches
  - the modeling may, however, differ with regard to the level of detail
  - in contrast to simulation, mathematical analysis usually necessitates making restrictive assumptions



Performance evaluation: analysis vs. simulation (continued)

- Advantages of mathematical analysis
  - The results can be obtained quickly
  - The results are exact
  - Gives insight
  - Allows optimization (though possibly difficult)
- Disadvantages of mathematical analysis
  - Requires restrictive assumptions
    => model can be too simple
    => performance evaluation of complex systems may be impossible
  - Even under simplifying assumptions the analysis may be difficult, calls for experts
  - The results may be limited to
    - equilibrium state
    - average values

S-38.3148 Simulation of data networks / Introduction

Performance evaluation: analysis vs. simulation (continued)

- Advantages of simulation
  - No constraints in the model building
    => allows performance evaluation of complex systems
  - Modeling is usually straight forward
- Disadvantages of simulation
  - Producing results is laborious (needs a lot of processing time)
  - Results are imprecise (though the precision can be increased by running more iterations)
  - Getting insight is more difficult
  - Optimization is more difficult (e.g., may be limited to the trial of a few parameter combinations)

Common pitfalls in simulation

- The goals of the simulation are not clearly defined
  - the system or important questions are not understood well enough
  - "let's simulate, the problem will become clearer "
- Not enough communication with the problem owner
- Inappropriate level of detail (too many / too few details)
  - with too many details any simulation program execution times will grow too much
  - with too few details (too abstract) model may require input data that is not available
- Focus is too much on the simulation program
  - as if the task is solely to write a demanding program
- Lack of expertise on
  - the system, performance analysis, statistical analysis
- Use of wrong tools
- Too much reliance on commercial simulators
  - which create the illusion that anybody can use the tool intelligently
  - misuse of animations

Common pitfalls in simulation (continued)

- Improper handling of stochastic features in the system
  - arbitrary or unjustified choice of distributions (e.g. the normal distribution)
- Reliance on one simulation
  - simulation must be repeated several times in order to assess the accuracy
  - when one single run is used, the statistical dependencies have to be accounted for
- Improperly handled initial phase
  - the initial part of the simulation (starts from a given state) is not representative and must be discarded
- Too short a simulation
- Poor random number generators
- Improper selection of seeds
  - random number streams (used for different purposes) in one simulation must not start with the same seed (e.g. 0) because this can create correlation between different processes
  - desirably the random number streams do not overlap at all

Theory

- Introduction to simulation
  - Simulation as a method for performance analysis
  - Classification of simulation models
  - Modeling, implementation and validation
- Flow of simulation -- generating process realizations
- Random number generation from given distribution
- Collection and analysis of simulation data
- Variance reduction techniques

System

- The system is a set of interacting entities
- The *state* of the system is a set of variables which at a given time instant define the system with a precision sufficient to the study
  - number of occupied trunks, number of packets in the buffer etc
- *Discrete system*
  - the values of the state variables take a set of discrete values (jump at discrete instants)
  - “digital” systems
- *Continuous systems*
  - the state variables change continuously as a function of time
  - the dependencies between them are often described by systems of differential equations
  - “analog” systems
- Often the use of discrete or continuous description of the system depends simply on the practicality of the approach (freedom of choice for the modeler)
  - depends on the time scale of events one is interested in
  - a discrete system can on a coarser level be considered as a continuous one (e.g. fluid queues)

Simulation models

- *Static* simulation model
  - relates to a given instant of time or the equilibrium state of the system -- time is not considered at all
  - used for estimation of some quantities from a given distribution
  - e.g. numerical integration of multi-dimensional integrals with the Monte Carlo method
- *Dynamic* simulation model
  - describes the development of the system in time

- *Deterministic* model
  - does not contain any stochastic elements
  - with given input the output is uniquely determined (possibly through very complicated relationships, but still…)
- *Stochastic* model
  - takes into account that in real world there is always some stochasticity (e.g. arrival instants and holding times of calls)
  - with given input the output is stochastic
  - by simulation one can assess the distributions or get estimates for expectations or other characteristic numbers of the distributions

S-38.3148 Simulation of data networks / Introduction

Simulation models (continued)

- There is some ambiguity in the use of the term “Monte Carlo simulation”
- Most often it refers to the *static* simulation
  - classical example is Buffon’s needle:
    - a needle with length d is cast on the floor with parallel lines at distances d
    - the probability that the needle in on a line is $2/\pi$
    - with repeated casts one gets an estimate of $2/\pi$ (and hence an estimate for π)
- Sometimes Monte Carlo -simulation refers to *stochastic* simulation in general (i.e. dynamic as well as static stochastic simulation)
  - this was the original use of the term by J. von Neumann and S. Ulam who during the World War II studied neutron transport in nuclear material
  - many of the ideas in simulation originate from this and related work

Simulation models (continued)

- Simulation of the (traffic) performance of telecommunication systems mostly use a simulation model which is
  - *discrete*
  - *dynamic*
  - *stochastic*

- In this course we are primarily interested in this kind of systems and simulation models

- With regard to the stochasticity there are two alternatives
  - one creates a model for the stochastic processes and generates the random variables from this known process
  - one uses measured sequences from real systems as stochastic input; this is called *trace-driven simulation*

Trace driven simulation

Advantages

- Credibility
  - real input
  - correct load conditions
- Allows easy verification of the model
  - compare with measured values
- Allows precise comparisons
  - as the load is exact (real) one can reliably compare the effect of even small differences in the system
- Less stochasticity
  - the trace is deterministic results are variable only due to stochasticity in other parts in the system

Disadvantages

- Complexity
  - requires a more accurate model
  - takes a lot of time
- Representativity
  - a trace taken from one system may not be representative for another system
- Only one point of comparison
  - a system which is optimized using one trace may not be optimal with another trace
- Limitedness
  - saving a trace takes a lot of space and one may need to restrict the trace to cover a relatively shot period of time
- Changing the load is difficult
  - one trace represents one load level
  - sensitivity studies are difficult or require several traces at different loads

Theory

- Introduction to simulation
  - Simulation as a method for performance analysis
  - Classification of simulation models
  - Modeling, implementation and validation
- Flow of simulation -- generating process realizations
- Random number generation from given distribution
- Collection and analysis of simulation data
- Variance reduction techniques

Steps in simulation of a stochastic system

- *Modeling* of the system as a time dependent stochastic process
- Generating realizations of the process
  - generating random numbers
  - event driven simulation
  - often the word simulation refers to this step only (from the point of view of traffic theory this is simulation in a restricted sense)
- Collection of data
  - transient phase vs. equilibrium
- Statistical analysis and conclusions
  - point estimators
  - confidence intervals

Creating a simulation model

- Simulation is intended to serve as a substitute for experiments in a real system
- Model must be as *correct* as possible
  - the conclusions drawn on the basis of the model must be the same as those drawn from measurements in a real system
- The model must be *credible*
  - only then can any model (even a correct one) have any impact on the decision making process
  - the worst case though is a model which is credible but erroneous

Principles in the model creation

- Define carefully what is going to be studied
  - which quantities have to be measured
  - what the model is to be used for
  - alternative system configurations which are to be compared
- Choose an appropriate level of detail
  - ask expert opinion for sensitivity assessment: which parts of the system are most critical from the point of view of the results
  - or use “coarse” simulation or analytical results for sensitivity analysis
  - incremental approach: start with a ”reasonable” level of detail, add more details as needed
  - don’t include details that are unimportant to the results
  - the level of detail must correspond to the level of input data

Model validation

- One has to make sure that the model actually describes the system
  - assumptions
  - the values or distributions of the input parameters
- Three different means for the verification
  - the intuition of experts
  - comparison with measurements in a real system
  - theoretical results

Model validation (continued)

- Expert intuition
  - a common and practical method
  - “brain storming” with people who know the system in order to define sensible assumptions and input data
  - an expert can easily recognize “impossible” results
- Comparison with measurements in the real system
  - the best and most reliable method
  - often difficult to apply: there is no real system, measurements are too expensive...
- Theoretical results
  - in some cases the system or some part of it can be analyzed under simplified assumptions
  - the simulation can be run under the same assumptions; at least then the results should match
  - this does not, however, guarantee that the model is generally valid

Building a simulation program

- Implementing and running a simulation program contains the steps mentioned on slide 22 except for model creation and conclusions, i.e.
  - generation of realizations for the stochastic process chosen as the model
  - collection of data
  - statistical analysis of the data
- Simulation program can be written using
  - a general purpose programming language, e.g. C tai C++,
  - some program libraries supporting simulation, e.g. CNCL
  - specific simulation languages (SIMULA, GPSS, MODSIM, SES/workbench, SIMSCRIPT)
  - tools developed for the simulation of specific systems
    - Communications network simulators: QualNet, OPNET, ns2, OMNeT++

General purpose languages vs. simulation languages

General purpose languages
(C, C++, Fortran...)

- Most users can use some of these languages
- Are available in most computers
- Code can be easily transported
- Low cost of the programs
- The code is often faster
- Flexibility, one can write “anything”
- Requires a lot of programming work
- Susceptible to errors -- requires careful verification of the code
- Special libraries can facilitate the work

Simulation languages (general purpose)
(SIMULA, GPSS, MODSIM, SES/workbench, SIMSCRIPT)

- Support many features needed in the programming of a simulation model
- Shorter development time -- lower total cost of the project
- Supporting features (or ready modules) for telecommunication system modeling)
- Can be used for simulation of other kind of systems, e.g. production systems
- Programming with the aid of the modeling constructs of the language

Simulation languages (telecom oriented)
(OPNET, QualNet, ns2, OMNET++)

- Developed specifically for the simulation of communication networks
- Contains “network building blocks”

Telecommunication network simulators

- Allow the simulation of networks of given type without actual programming
- The type of the network (or network element) is picked from a menu
- Typical elements are, e.g.
  - LAN-type (Ethernet, token ring...)
  - Interconnection elements of LANs (bridges, routers...)
  - Stations attached to the LAN (PCs, workstations...)
  - Also, transport layer protocols are implemented (such as TCP)
- Fast implementation
  - e.g., for testing different scenarios
- Easy to produce graphical representations
- Stiff, limited to the available elements
- Can easily lead to misuse of simulation

Planning and verification of the program

- Use top-down planning
- Modular structure of the program
  - divide the program in clearly defined blocks which communicate through well-defined interfaces
- Object oriented programming
  - use generic object classes (system components)
  - using the well tested generic classes one can create more specific classes for own purposes
- Debug using tracers
- Let others read the program
- Make test runs with different input parameters
  - check the sensibility of the results in each case
- Make test runs in simplifies cases
  - where the correct results are known or can be easily calculated
- Animated results may make possible errors apparent