Simulations and Tools for Telecommunications

521365S:

OPNET - Network Simulator

Jarmo Prokkola
Research team leader, M. Sc. (Tech.)

VTT Technical Research Centre of Finland
Kaitoväylä 1, Oulu
P.O. Box 1100, 90571 Oulu, Finland
GSM: +358 40 706 1549
Email: jarmo.prokkola@vtt.fi

22.04.2008: Tietotalo, University of Oulu
About the presenter (me)

- Research scientist / research team leader in VTT Technical Research Centre of Finland from 2004 –
- Research scientist in University of Oulu, Centre for Wireless Communications & Telecommunication laboratory from 1998 to 2004
- Research interests include:
  - Cross-layer design methods
  - Wireless Networks
  - Channel Access methods
  - Ad Hoc networks
  - MAC protocols
  - Quality of Service (QoS) in heterogeneous networks
  - QoS measurement methods and tools
- Working with OPNET since 1999
  - Mainly wireless system research and ad hoc networking techniques
Contents

• General information and Introduction
• OPNET modeling hierarchy
  • Network Domain
  • Node Domain
  • Process Domain
• Simulations & analysis
• External System Domain
• Physical layer modeling (radio link)
• Packet formats
• OPNET products & additional modules
• Tips for the rookies
• Summary
General Information

- A common presentation of the OPNET simulator (OPNET Modeler) is provided
- OPNET is extensive and powerful simulation software with wide variety of possibilities
  - Enables the possibility to simulate entire heterogeneous networks with various protocols
- Development work was started in 1986 by MIL3, Inc., while nowadays the company is called OPNET Technologies, Inc.
  - In 2007: 490 employees, revenue: ~ 95 M$
- Originally the software was developed for the needs of military, but it has grown to be a world leading commercial network simulation tool
- OPNET is quite expensive for commercial usage but there are also free licenses for educational purposes
Introduction

• OPNET is a high level **event based network level simulation tool**
  • Simulation operates at “packet-level”
  • Originally built for the simulation of fixed networks
    • OPNET contains a huge library of accurate models of commercially available fixed network hardware and protocols
  • Nowadays, the possibilities for wireless network simulations are also very wide
    • Accurate radio transmission pipeline stage for modeling of the physical layer (radio interface)
    • The simulator has a lot of potential, but there exists typically a lack of the recent wireless systems
      • Much of the work considering new technologies must be done by oneself
  • OPNET can be used as a research tool or as a network design/analysis tool (end user)
    • The threshold for the usage is high for the developer, but low for the end user
The structure of OPNET

• OPNET consists of high level user interface, which is constructed from C and C++ source code blocks with a huge library of OPNET specific functions

• Hierarchical structure: modeling is divided into three main domains:
  • Network domain
    • Networks + sub-networks, network topologies, geographical coordinates, mobility
  • Node domain
    • Single network nodes (e.g., routers, workstations, mobile devices, …)
  • Process domain
    • Single modules and source code inside network nodes (e.g., data traffic source model, IP protocol, …)
• With OPNET it is also possible to run external code components (External System Domain, ESD)
The Various Tools of OPNET

- Network model editor
- Node model editor
- Process model editor
- Source code editing environment
- Antenna pattern editor
- Modulation curve editor (SNR – BER behavior)
- Packet format editor
- Analysis configuration tool
- Simulation tool
- ICI editor (Interface Control Information)
- Probe model tool (organization of result collection)
- Link model editor (properties of fixed link models)
- Path model editor (for routing and modeling virtual circuits)
- Demand model editor (wide scale application modeling)

- OPNET Animation viewer
The Network Domain (1/4) – Global Network Modeling

- Network level (main window)
The Network Domain (2/4)

- Network domain specifies the overall scope of the system to be simulated.
- It is a high-level description of the objects contained in the system.
- Network model specifies the objects in the system as well as their physical locations, interconnections, and configurations.
The Network Domain (3/4)

sub-networks

• An example of sub-networking (WLAN)
The Network Domain (4/4) – Mobility

- OPNET supports wireless mobile nodes
- Also, satellite modeling is an inbuilt feature of OPNET
- Mobility can be realized at least in three different ways

- **Mobility trajectory**
  - A node follows a predetermined trajectory during simulation (drawn or defined step by step).

- **Mobility vector**
  - A node moves according to a mobility vector defined with node attributes, which can be modified during simulation.

- **Manipulation of node’s coordinates**
  - A processor module is created, which directly modifies node’s coordinates during simulation according to the specified model.
Node Domain (1/3) – Individual Network Node Modeling

- Example: From network to node domain (WLAN workstation)
Node Domain (2/3)

- The node model specifies the internal structure of a network node.
- Typical nodes include workstations, packet switches, satellite terminals, remote sensors.
- Nodes can be fixed, mobile, or satellite type.
- A node can also be a special kind of node representing e.g., an entire Ethernet network and its aggregate traffic as one entity.
Notice the structure (OSI reference model):

- Application layer
- Presentation & session layers (Application interface)
- Transport layer (TCP & UDP)
- Network layer (IP)
- Link layer (ARP, WLAN-MAC)
- Physical layer (receiver, transmitter)
Process Domain (1/5) – Modeling Single Processor Entities, Algorithms, Protocols, etc.

Example: from node domain to process domain (TCP processor)
Process Domain (2/5)

- Process models are used to specify the behavior of processor or queue modules, which exists in the Node Domain.
- A module is modeled as a finite state machine (FSM).
- FSM consists of states with transitions and conditions between them.
Process Domain (3/5) – The Source Code

- States consist of OPNET flavored C or C++ code

- An example of “the code level”

```c
/* Get the received packet. */
ev_ptr->pk_ptr = op_pk_get (intcpt_stream);
IT (ev_ptr->pk_ptr == OPC_NIL)
{
lw_level_error = OPC_TRUE;
op_prg_log_entry_write ([1], loghdl), "TCP SEQ_RCV failed - unable to get packet from input stream.
}
else
{
/* Determine the socket addresses for the packet. */
if ((op Educación_get (cli_ptr, "src_addr", addr_ptr)) != OPC_COMP_CODE_FAIL
       (op_pk_nf_addr = ev_ptr->pk_ptr, "fields", &pk_fd_ptr) == OPC_COMP_CODE_FAIL

   lw_level_error = OPC_TRUE;
op_prg_log_entry_write ([1], loghdl), "TCP SEQ_RCV failed - unable to obtain addressing information
rem_add = INETC_ADDRESS_INVALID;
}
else
{
/* Store the address and port information locally. */
rem_add = *addr_ptr;
/* Check that the address is valid. */
if (inet_address_equal (*addr_ptr, INETC_ADDRESS_INVALID) == OPC_TRUE)
   lw_level_error = OPC_TRUE;
if (tcp_trace_active)
   op_prg_pdo_print_major ("Received packet with invalid source",
   else
   rem_port = pk_fd_ptr->src_port;
local_port = pk_fd_ptr->dest_port;
/* Determine the local and remote TCP connection. */
/* process "key" identifiers (maintained locally). */
local_key = pk_fd_ptr->remote_key;
rem_key = pk_fd_ptr->local_key;
```

22.04.2008, Jarmo Prokkola
Process domain (4/5) – Source Code Editor

- Basically, process model editor is just a tool to ease the development of C or C++ source code of the desired model.
Process Domain (5/5) – Attributes

- Example: the parameters of 802.11 WLAN MAC
- An easy way for the user to modify model attributes
- “Promote” functionality enables the attribute modification in the upper domains
Simulations (1/3) - Simulation tool

- With OPNET’s simulation tool, it is possible to combine several low level attributes and make series of simulation iterations.
The basic simulations with OPNET are done as a function of simulation time
- Accurate network behavior
- The level of event accuracy can be extended to be as detailed as needed

Simulation results as a function of time are typically as such not suitable as scientific results, since statistical accuracy is needed
- A certain situation can be first verified with a simple simulation run, but then several runs should be done with different random generator seed values.
- The typical scientific simulation results are graphs of average statistics based on several simulation iterations. As such, the OPNET’s basic analysis tool is not the best tool for drawing graphs like this, but it can be used to collect and export statistics to a third party software.
Simulations (3/3) – A Result Example

- **BC-MAC (NSCD) with BCCA clearly outperforms the traditional IEEE 802.11**
  - Consider e.g., if an application desires packet losses below $10^{-3}$: BC-MAC can offer over 20 times more capacity than 802.11!
Analysis Configuration tool

- A tool for plotting and analyzing the simulated results
- The tool is practical for quick analysis but is maybe not the best one for making publishable figures
- Provides also a possibility to write the selected and filtered results to a text file for further analysis (i.e., enables the possibility to use an external analysis tool)
External System Domain (ESD)

An external system is OPNET's representation of a model whose behavior is determined by a code external to OPNET.

Such a model can be anything from a microchip to a model of user behavior pattern.

OPNET passes data to external system and receives data from it with no implicit knowledge of how the external code processed the data.
On the Physical Layer Modeling (radio link) (1/3)

The physical layer is modeled with pipeline stages, which are used to calculate step by step the total effect of the physical transmission medium including all the interference caused by other users.

- Each pipeline stage is a model made with OPNET flavored C or C++.
On the Physical Layer Modeling (radio link) (2/3)

- The used pipeline stage models can be defined by transmitter and receiver attributes.
- General channel settings can be also determined with these attributes. The attributes can be also modified during the simulation by the processors.
On the Physical Layer Modeling (radio link) (3/3)

- Modulation curves can be edited with a special editor.
Radio link: Modulation curves

- Modulation curves specify the average SNR-BER behavior of the received packet segment.
- SNR (or Eb/No) in OPNET is specified as received signal strength divided by received noise + total interference power.
- Modulation curves can be simulated with external simulator (e.g., Matlab) and imported to model physical layer behavior accurately when needed special situations (fading channel, channel coding, interference suppression...).
Specific antenna patterns can be defined
Example: UMTS station
Radio link: Antenna patterns (2/2)

- Antenna pattern editor can be used to determine specific antenna patterns.
- Basically, each receiver and transmitter channel can be connected to antennas, which can have different patterns.
- Antennas can be dynamically directed by processor modules during a simulation.
- Antenna pattern tool is somewhat clumsy to use, but patterns can also be made with EMA-code (External Model Access) functionality.
Packet format editor

- Packet formats define the internal structure of packets with a set of fields
- Illustrative and valuable tool in OPNET, since the whole operation of the simulator is structured from packet basis

- An example of WLAN MAC packet
OPNET Products

Network R&D
- OPNET Modeler – The main simulation & development tool
  + Wireless suite and Defense additions

Capacity management
- IT Guru (network and system capacity planning for enterprises)
- SP Guru (network planning and engineering for service providers)

Network operations
- IT Sentinel (network audit, security, and policy-compliance for enterprises)
- SP Sentinel (network audit, security, and policy-compliance for service providers.)
- IT and ST Netcop (unified network views and situational awareness)

Application performance management
- ACE Analyst (analytics for networked applications)
- ACE Live (end-user experience monitoring & real-time network analytics)
- SLA Commander (application service level monitoring)
- OPNET Panorama (real-time application monitoring and analytics)
Additional Modules and Features for Modeler

- **UMTS** (WCDMA)
- **WiMAX** (802.16-2004, 802.16e-2005)
- **DOCSIS** (Data Over Cable Service Interface Specification)
- **IPv6 + Mobile IPv6**
- **MPLS** (Multi-Protocol Label Switching)
- **PNNI** (Private Network-Network Interface)
- **Terrain Modeling**
  - **TIREM** (propagation modeling calculation)
- **Satellite Capability** (dozens of models for satellite systems)
- **Joint Communications Simulation System (JCSS)** (Military orientated models)
- **Vendor Devices** (Dozens of vendor specific devices)
- **Integration with 3rd Party Products** (several modules)
Finally, for the rookies

- OPNET tutorial is a good starting point for beginners
- Various tutorials guide the user through the basic functionalities of OPNET
- Wide product documentation is available to help the user through all the issues considering OPNET
- Especially discrete event simulation API section gives valuable information about the function libraries of OPNET
Summary

- OPNET is event based (discrete time) network simulation tool
- The software is powerful but also demanding at least for the developers
  - Using only high-level simulations is quite straightforward
- Modeling is done mainly in three levels
  - Network Domain (top level)
    - Upper levels hide the complex structure of the lower level components, which can be totally invisible to the end user
  - Node Domain
  - Process Domain
- Different layers and functionalities can practically be modeled as accurately as needed with EMA and ESD functionalities
- OPNET includes various additional modules and tools for easing the usage of the software
References

• OPNET Modeler + Wireless (version 14.0 (10.0))
• OPNET (versions 14.0 (10.0)) Documentation
• OPNET Technologies, Inc. WWW-page, http://www.opnet.com