Orion: Interconnect model/simulator

Ramon Canal
CTD – Master CANS

Departament d'Arquitectura de Computadors
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Introduction

- Interconnection networks are becoming an import part of micro-processing
- System power consumption is increasingly becoming at least equally as important as performance
- Interconnect networks are consuming an ever greater percentage of system power

Introduction (cont.)

- Interconnection networks are seen as only scalable solution to inter-processor comm by single-chip multiprocessors
- Soon, routers and links will be critical components of a microprocessor system
- InfiniBand switch is estimated to dissipate almost 37.5% of the blade's power budget in a Mellanox server blade

Introduction (cont.)

- Simulator is constructed within the Liberty Simulation Environment (LSE)
- Goal: provide a complete platform for exploring interconnected microprocessors, whether single-chip or spanning multiple chips, at the architectural level
- Power model for interconnects usable from other tools

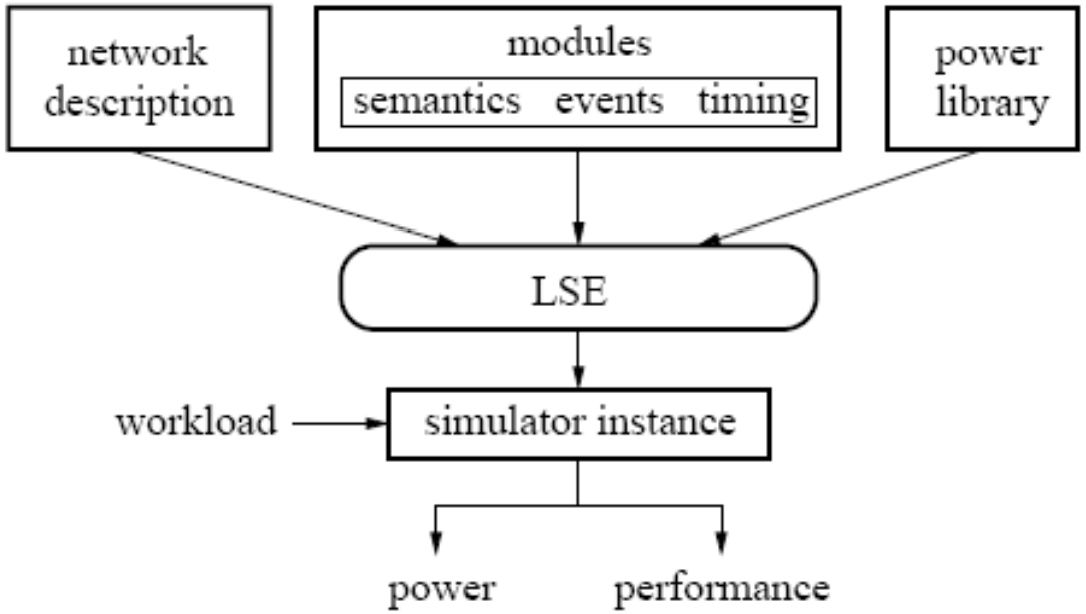
Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Simulator Infrastructure

- Adopted LSE as basic simulator infrastructure
- LSE targets fast design space exploration for modern microprocessors
- LSE models physical hardware blocks as logical functional models which communicate through ports
- Integration of power models is based on the event subsystem of LSE that facilitates collection of execution stats
- Power models in power simulation are hooked to these events
- Specific power model calculates and accumulates the energy which is consumed

Process of building a simulator in LSE



Building blocks of interconnection network

- Interconnection networks can be decomposed into component modules
- Modules are parameterized so that they can be reused
- A relatively small library of modules is able to represent an extensive range of architecture choices

Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Power Modeling

- Architectural-level parameterized power models are derived for several major building blocks (FIFO buffers, crossbars and arbiters)

Component Power Modeling

- For each component, first the canonical structure is described in terms of architectural and technological parameters
- The detailed analysis is performed to determine parameterized capacitance equations
- Capacitance equations and switch activity estimation are combined to determine energy consumption/component operation

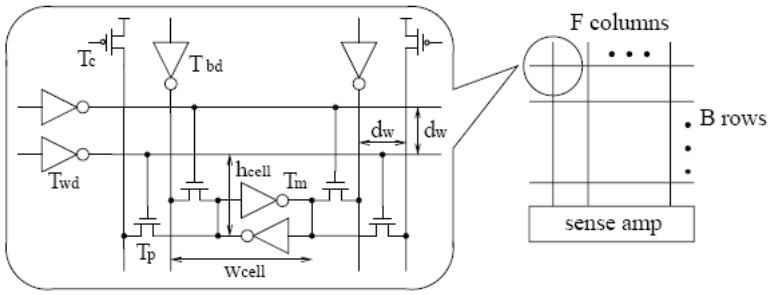
Power Model for FIFO buffer

Table 1. Terminology

$C_g(T)$	gate capacitance of transistor/gate T
$C_d(T)$	diffusion capacitance of transistor/gate T
$C_a(T)$	$C_g(T) + C_d(T)$
$C_w(L)$	capacitance of metal wire of length L
E_x	$\frac{1}{2}C_x V_{dd}^2$ or $C_x V_{dd}^2$ depending on how to count switches, provided C_x is defined

Table 2. Model for FIFO buffers

Canonical structure
A FIFO buffer with 1 read port and 1 write port



Architectural parameters	
B	buffer size in flits
F	flit size in bits
P_r	number of buffer read ports
P_w	number of buffer write ports

Technological parameters	
h_{cell}	memory cell height
w_{cell}	memory cell width
d_w	wire spacing

Capacitance equations	
wordline length	$L_{wl} = F(w_{cell}+2(P_r+P_w)d_w)$
bitline length	$L_{bl} = B(h_{cell}+(P_r+P_w)d_w)$
wordline cap.	$C_{wl} = 2FC_g(T_p)+C_a(T_{wd})+C_w(L_{wl})^\star$
read bitline cap.	$C_{br} = BC_d(T_p) + C_d(T_c) + C_w(L_{bl})$
write bitline cap.	$C_{bw} = BC_d(T_p) + C_a(T_{bd}) + C_w(L_{bl})$
precharge cap.	$C_{chg} = C_g(T_c)$
memory cell cap.	$C_{cell} = 2(P_r + P_w)C_d(T_p) + 2C_a(T_m)$
sense amp energy	E_{amp} from empirical model [28]

Operation energy equations	
δ_{bw}	number of switching write bitlines
δ_{bc}	number of switching memory cells
read energy	$E_{read} = E_{wl} + F(E_{br} + 2E_{chg} + E_{amp})$
write energy	$E_{wrt} = E_{wl} + \delta_{bw}E_{bw} + \delta_{bc}E_{cell}$

($\star$) T_p is the pass transistor connecting bitlines and memory cells, T_{wd} is the wordline driver, T_{bd} is the write bitline driver, T_c is the read bitline precharge transistor, T_m is the memory cell inverter.

Discussion

- Power models are based on detailed estimates of gate and wire capacitance and switching activity
- In process of validating power models against measured power numbers of existing routers and against low-level power estimation tools

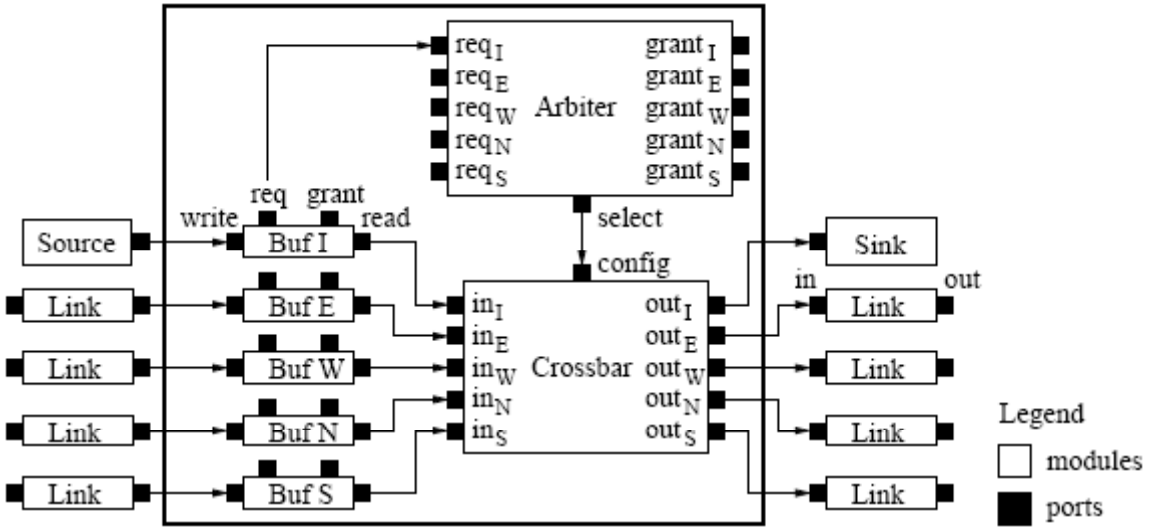
Discussion (cont.)

- Currently working with chip-to-chip and on-chip link designers to develop parameterized link models
- Power models (coded in C) as part of Orion's release

Walkthrough example: wormhole router

- Will move a head flit (smallest unit of flow control, and is a fixed-sized unit of a packet)
- Assumptions:
  - router has 5 input/output ports
  - 4 flit buffers per input port
  - each flit 32 bits wide
  - 5x5 crossbar
  - 4:1 arbiter per output port

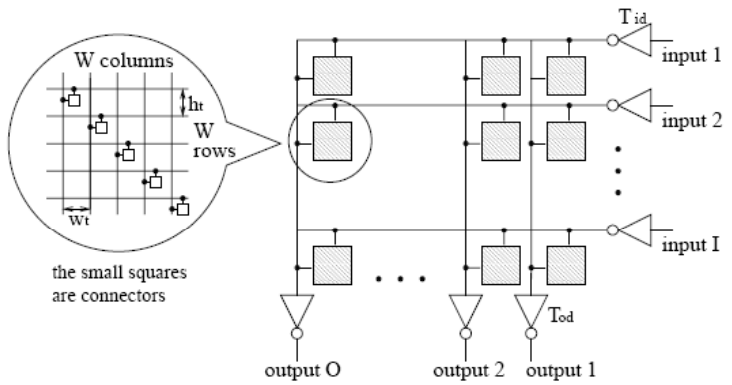
Model of simple wormhole router in Orion



$$E_{flit} = E_{wrt} + E_{arb} + E_{read} + E_{xb} + E_{link}$$

More Power Models

Table 3. Model for matrix crossbars

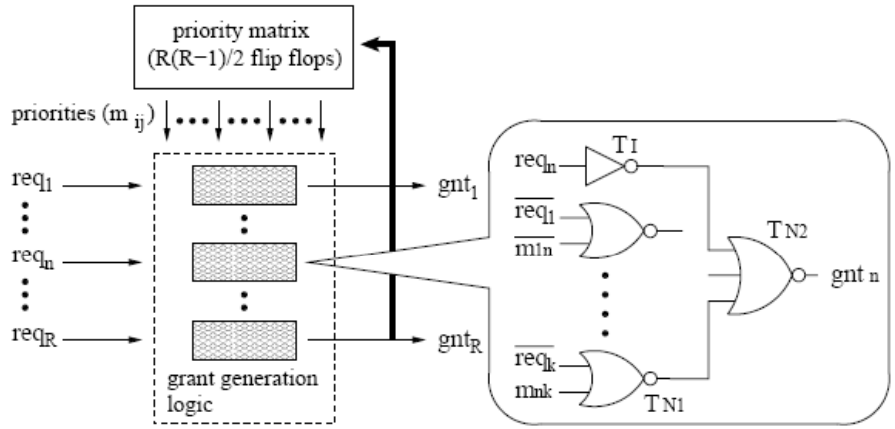
Canonical structure and notation	
	
connector	either a tri-state buffer or a transmission gate
C_{in_cnt}	input node cap. of a connector
C_{out_cnt}	output node cap. of a connector
C_{ctr_cnt}	control node cap. of a connector
Architectural parameters	
I	number of crossbar input ports
O	number of crossbar output ports
W	port width in bits
Technological parameters	
h_t	track height
w_t	track width

Capacitance equations	
input line length	$L_{in} = O \cdot W \cdot w_t$
output line length	$L_{out} = I \cdot W \cdot h_t$
input line cap.	$C_{xb\_in} = O \cdot C_{in\_cnt} + C_a(T_{id}) + C_w(L_{in})^*$
output line cap.	$C_{xb\_out} = I \cdot C_{out\_cnt} + C_a(T_{od}) + C_w(L_{out})$
control line cap.	$C_{xb\_ctr} = W \cdot C_{ctr\_cnt} + C_w(\frac{L_{in}}{2})$
Operation energy equations	
δ_{xi}	number of switching input bits
δ_{xo}	number of switching output bits
traversal energy	$E_{xb} = \delta_{xi} E_{xb\_in} + \delta_{xo} E_{xb\_out}$

$(*)$ T_{id} is the input driver, T_{od} is the output driver.

More Power Models (cont.)

Table 4. Model for matrix arbiters

Canonical structure and notation


C_{FF}	switch capacitance of a flip flop
C_{FC}	clock capacitance of a flip flop

Architectural parameters	
R	number of requesters

Capacitance equations	
request cap.	$C_{req} = C_a(T_I) + (R-1)C_g(T_{N1}) + C_g(T_{N2})^*$
grant cap.	$C_{gnt} = C_d(T_{N2})$
priority cap.	$C_{pri} = C_{FF} + 2C_g(T_{N1})$
internal cap.	$C_{int} = C_d(T_{N1}) + C_g(T_{N2})$
clock cap.	$C_{clk} = C_{FC}$

Operation energy equations	
δ_{ar}	number of switching request signals
δ_{ap}	number of switching priority bits
δ_{ai}	number of switching internal nodes
arbitration energy	$E_{arb} = \delta_{ar}E_{req} + \delta_{ap}E_{pri} + \delta_{ai}E_{int} + (E_{gnt} + E_{xb\_ctr})$

(*) T_{N1} is the first level NOR gate, T_{N2} is the second level NOR gate, T_I is the inverter.

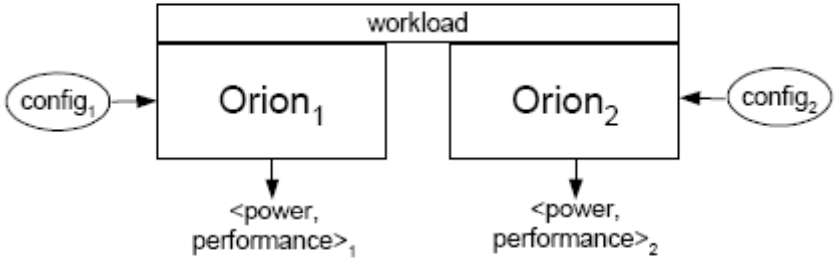
Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

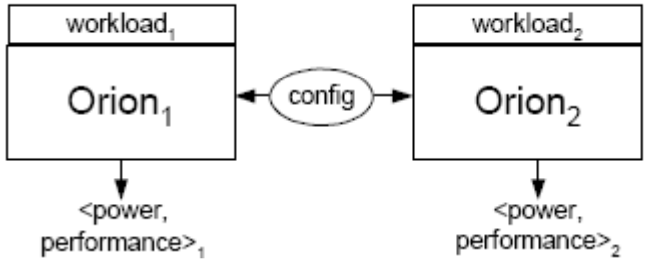
Case Studies

- Authors envision three primary uses for Orion
- They provide examples for each one of the envisioned uses
- The case studies are various design space explorations over the same system

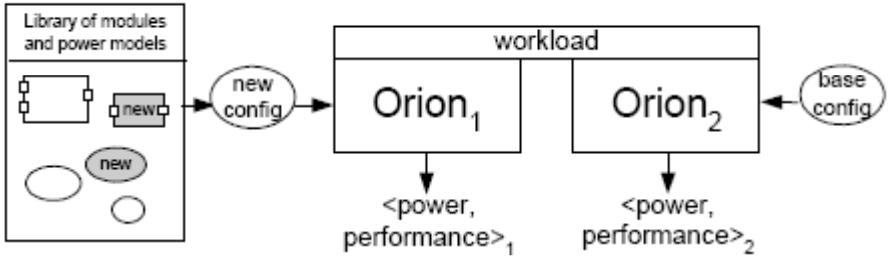
Three potential ways of utilizing Orion



(a) Exploring different configurations



(b) Exploring different workloads



(c) Exploring new microarchitectures

Experimental Setup

- 16-node network
- Each router has 5 physical bidirectional ports
- Router keeps count of available buffers
- No dropped packets
- Source dimension-ordered routing is used
- Simulator generates uniformly distributed traffic to random destinations unless otherwise mentioned

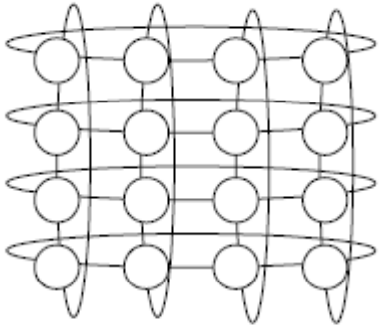
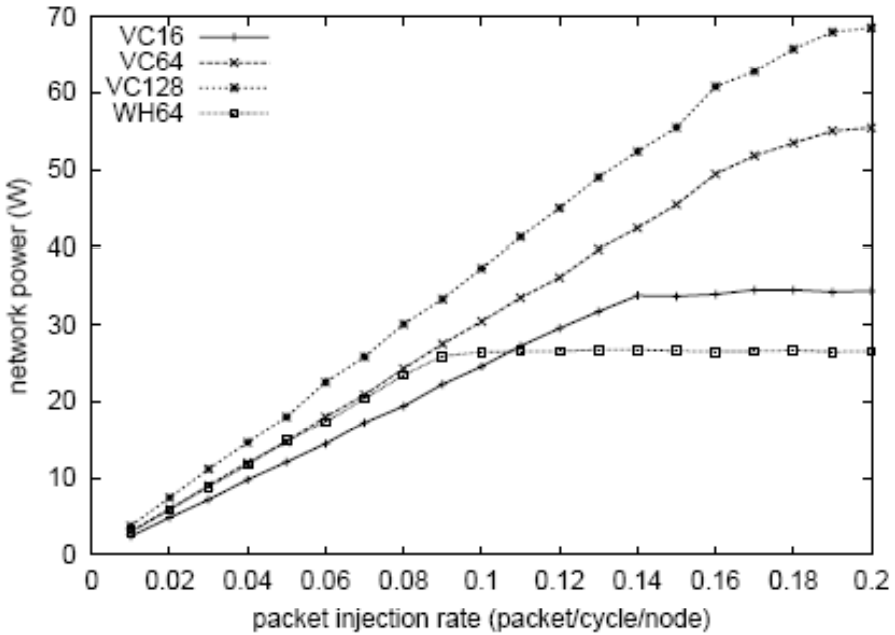


Figure 4. A 4-by-4 torus network.

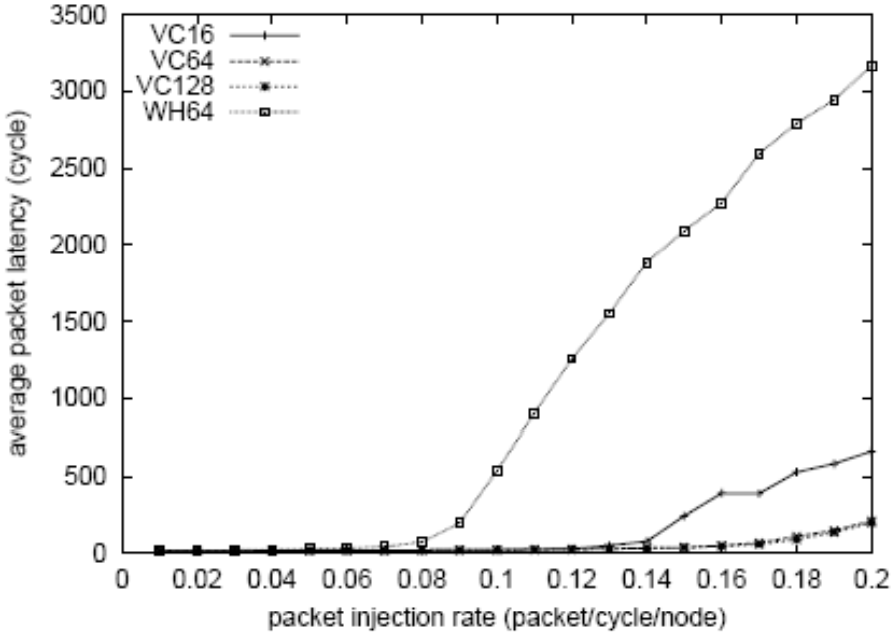
Exploring Configurations: wormhole vs. virtual-channel

- Four different router configurations are compared:
  - wormhole router with 64-flit input buffer per port (WH64).
  - virtual-channel (VC) router with 2 VCs per port and 8-flit input buffer per VC (VC16).
  - virtual-channel router with 8 VCs per port and 8-flit input buffer per VC (VC64).
  - virtual-channel router with 8 VCs per port and 16-flit input buffer per VC (VC128).

Results Case 1

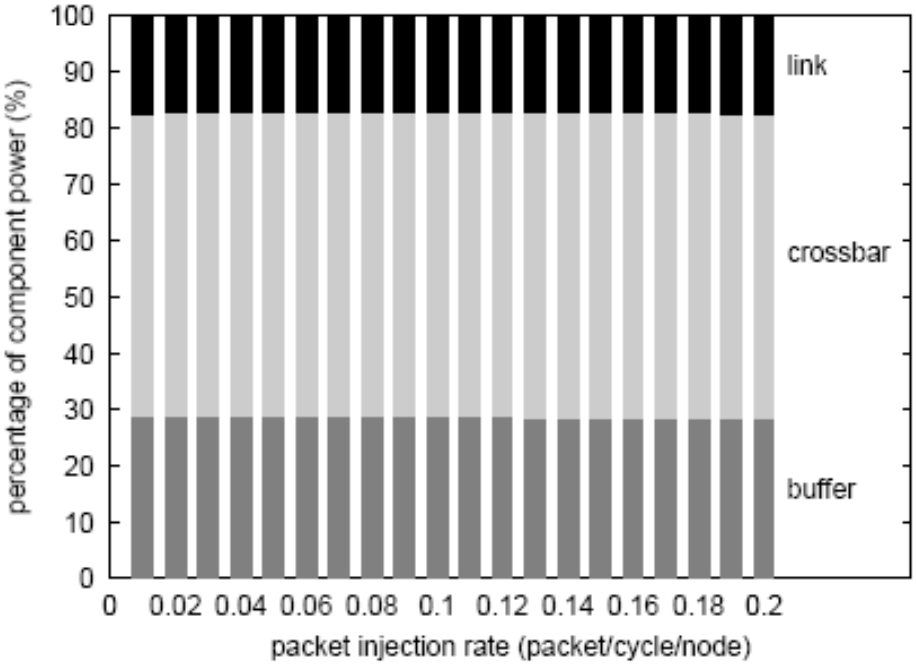


(b) total network power



(a) average packet latency

Results Case 1

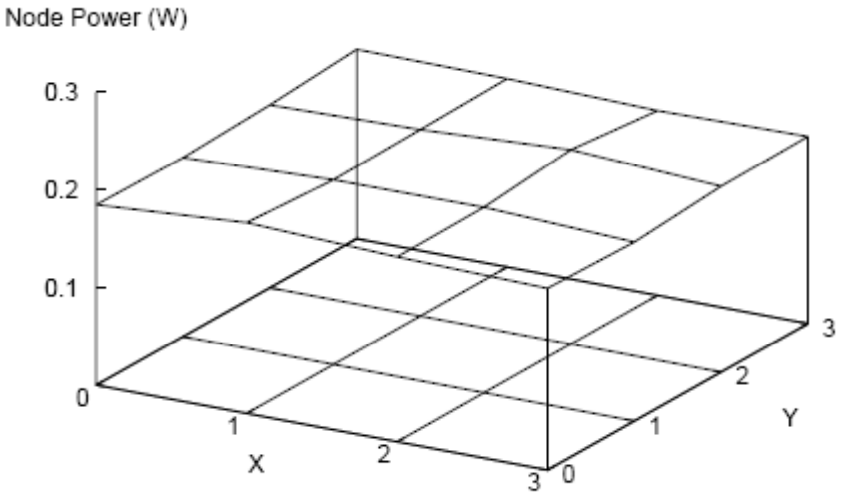


(c) VC64 average power breakdown, arbiter power is invisible at current scale.

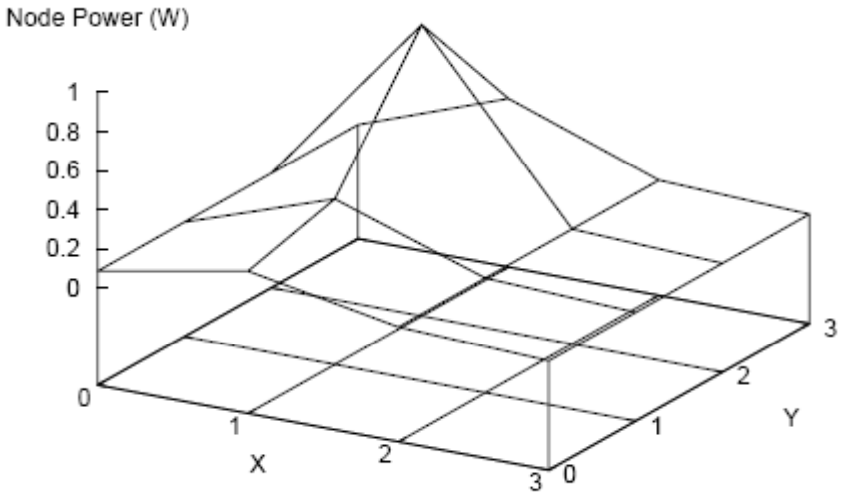
Exploring Workloads: broadcast vs. uniform traffic

- Compares two traffic patterns
  - uniform random traffic, *i.e.* each node injects packets to randomly distributed destinations other than itself in the network.
  - broadcast traffic, *i.e.* one node injects packets to all the other nodes in the network.

Results Case 2



(a) uniform random traffic

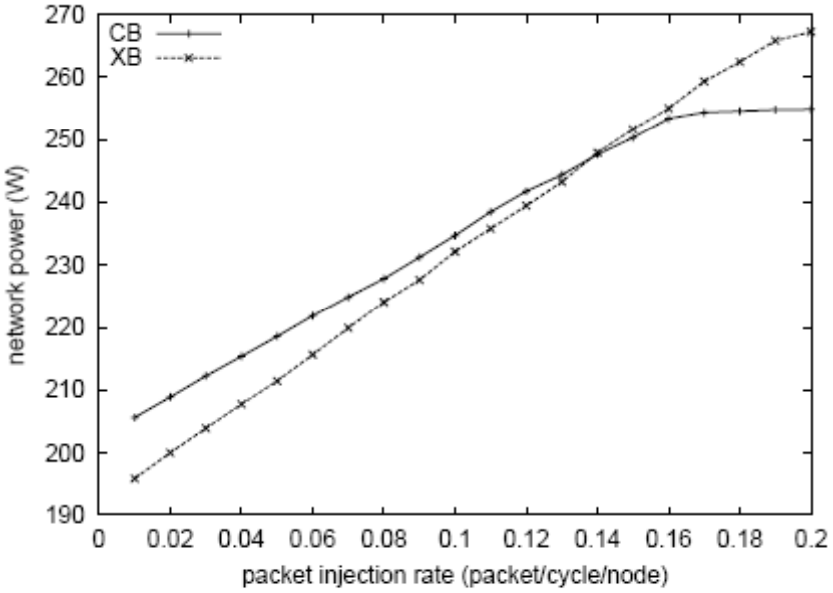


(b) broadcast traffic from node (1,2)

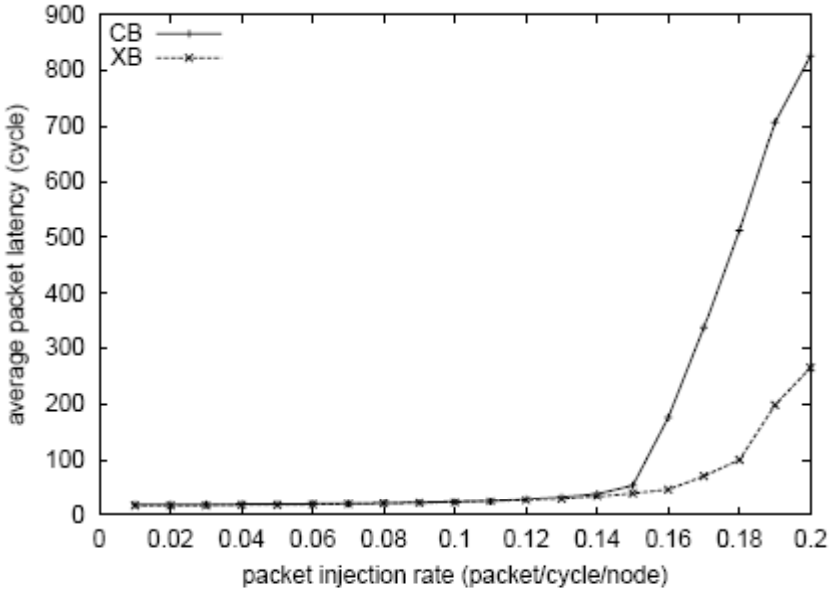
Exploring new micro-arch: central buffered routers

- Two different router architectures are explored:
  - Central-buffered router with a 4-bank central buffer, each 1 flit wide, 2560 chunks (2560 rows, each row 4-flit wide), 2 read ports, 2 write ports, and a 64-flit input buffer at each port (CB).
  - Input-buffered crossbar-based router with 16 virtual channels, 268-flit input buffer per VC and a 5x5 crossbar(XB).

Results Case 3 (random traffic)

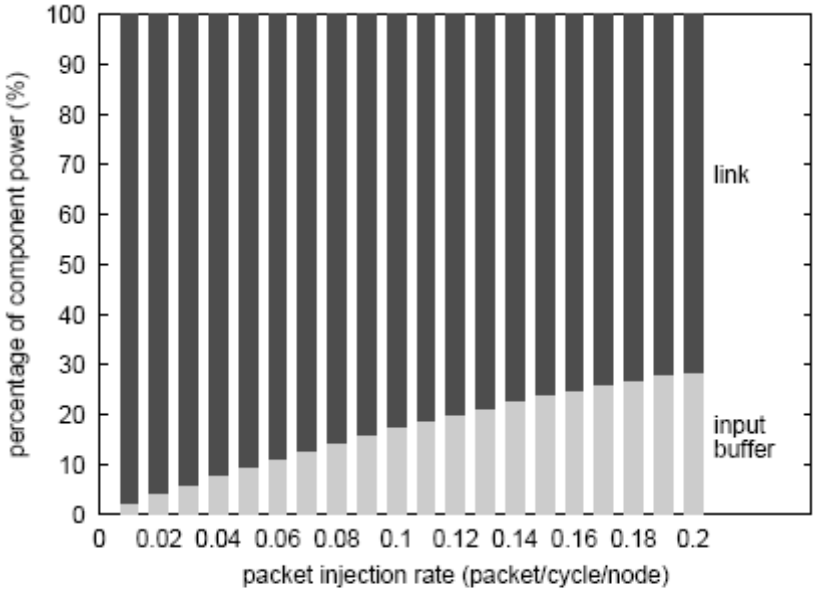


(b) total network power (random traffic)



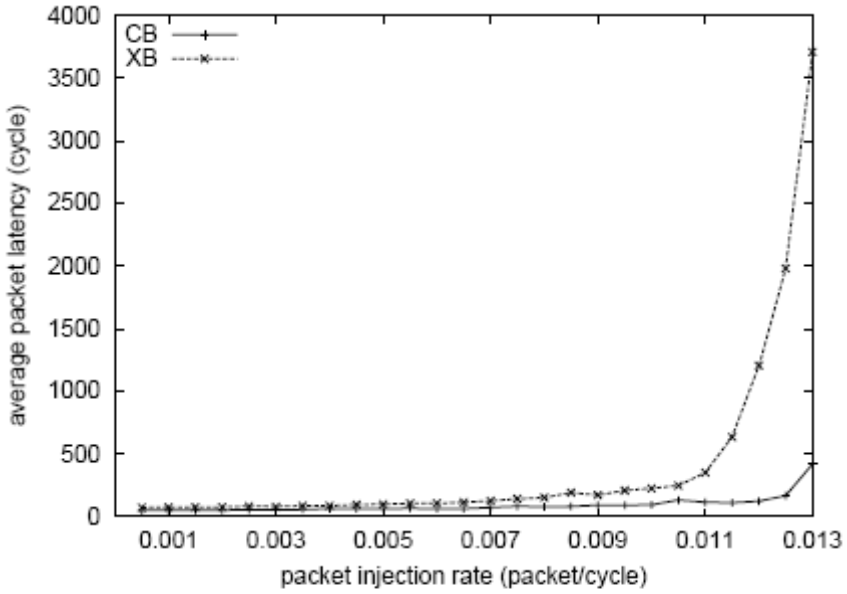
(a) average packet latency (random traffic)

Results Case 3 (random traffic)

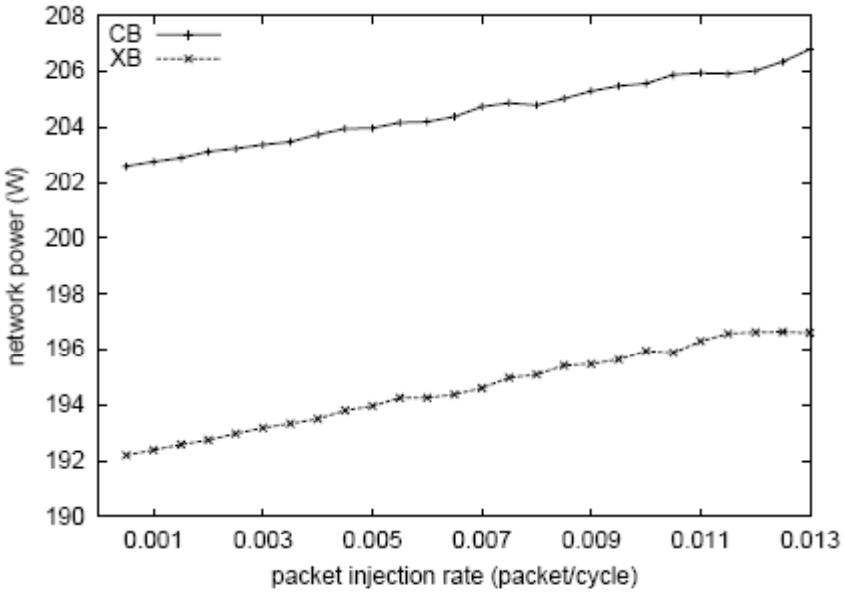


(c) average power breakdown of XB router (random traffic), arbiter power and crossbar power are invisible at current scale.

Results Case 3 (broadcast traffic)

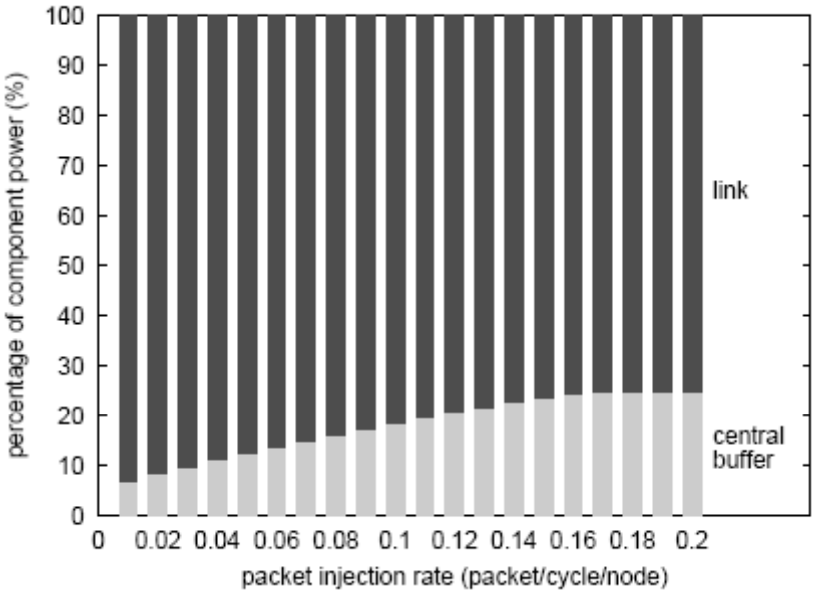


(d) average packet latency (broadcast traffic)



(e) total network power (broadcast traffic)

Results Case 3 (broadcast traffic)



(f) average power breakdown of CB router (random traffic), arbiter power and input buffer power are invisible at current scale.

Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Related Work

- Architecture-level mechanisms for power-efficient interconnection networks are sorely lacking
- Low-level tools require complete RTL level code and take on the order of hours
- Previous studies for power models focused on power consumption of different network topologies
- Models were adopted based on transistor count, switch width or low-level power estimation tool estimates

Agenda

- Introduction
- Dynamic Network Simulator
- Power Modeling
- Case Studies
- Related Work
- Conclusions

Conclusions

- Orion is a useful tool/model for rapidly exploring power-performance tradeoffs in network micro-architecure design