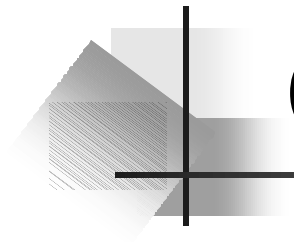
CommBench

--- A Telecommunications Benchmark For Network Processors

ECE 697J Active Networks

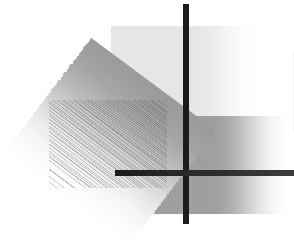
Presenter: Jianhong Xia

11/14/2002



Outline

- Motivation
- CommBench
- Data & Tools
- Characteristics
- Examples
- Summary



Motivation

- New requirements push network processors
- How to evaluate and select chips
- SPEC is designed for workstations/PCs
  - not focusing on defined I/O
  - not considering packet size
  - not considering performance
- CommBench
  - streaming data flow
  - packet-based processing task

CommBench

- Benchmark Applications
  - HPA (Header-Processing Applications)
    - per-packet basis
    - independent of size and type of the packet payload
    - RTR, FRAG, DRR, TCP
  - PPA (Payload-Processing Applications)
    - access/modify the content of a packet
    - typically executed on a stream of packets
    - CAST, ZIP, REED, JPEG

Data & Tools

- All benchmark programs
  - Run on SUN UltraSparc II under SunOS 5.7
  - Compiler is gcc 2.8.1 (O2)
  - Tools
    - **Shade** and **SpixTools** for instruction mix
    - **Dinero** for cache simulation
- Data types for different applications
    - HTML data (plain text)
    - Binary program code
    - JPEG coded image data

Benchmark Characteristics

- Compare to SPEC
  - Code and kernel size
  - Computational complexity
  - Instruction frequency
  - Cache performance

Characteristics --- Code and Kernel Size

- Static code size
  - an order of magnitude smaller than SPEC

Benchmark	Code Size C lines	Code size object bytes
CommBench	5,750	97,000
SPEC	48,700	678,000

- Dynamic kernel
  - much smaller than SPEC

Benchmark	Instr. at least once	Instr. for 99%	Instr. for 90%
CommBench	3,430	500	275
SPEC	35,700	9,700	3,390

Characteristics --- Computational Complexity

- Computational Complexity
  - Respect to the number and size of processed packets
  - Based on the # of instructions
- Definition of $N_{a,l}$
  - # of instructions per byte for application a on a packet of length l

HPA a	**N** a,64	**N** a,576	**N** a,1536
TCP	10.3	1.2	0.4
FRAG	7.7	0.9	0.3
DRR	4.1	0.5	0.2
RTR	2.1	0.2	0.1

PPA a	**N** a,¥ **(enc)**	**N** a,¥ **(dec)**
REED	603	1052
ZIP	226	35
CAST	104	104
JPEG	81	60

Characteristics --- Instruction Frequency

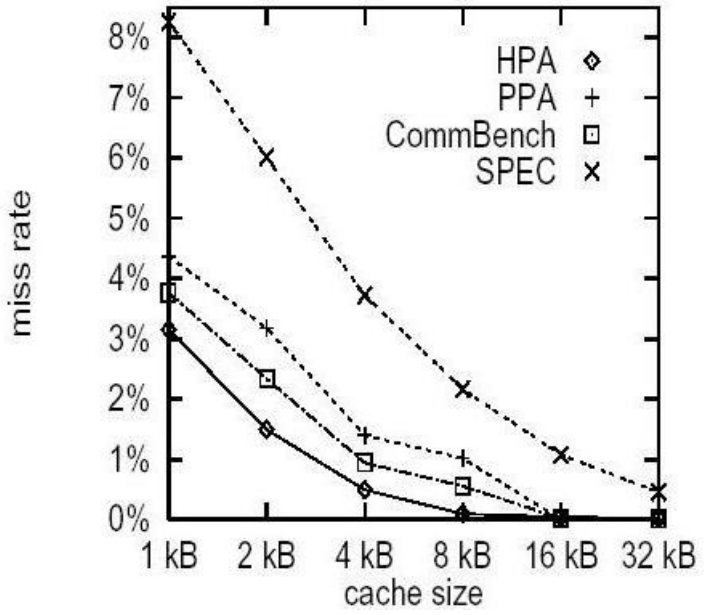
- Instruction Frequency
  - Almost the same as SPEC (not shown here)

HPA	%	PPA	%
load	27	add/sub	22
cond. branch	18	load	18
compare	18	cond. branch	13
add/sub	13	shift	13
store	6	compare	12
logic	6	logic	11
shift	4	store	7
load imm.	2	load imm.	1
jmpl	1	save/restore	1

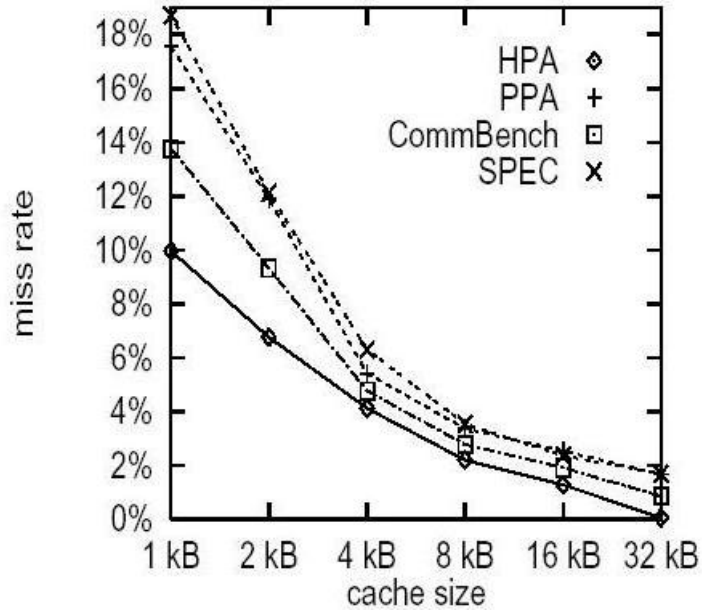
- For CommBench only
  - Instruction mixes are different for HPA & PPA
  - Indicate that network processors must deal with both streaming and header processing applications

Characteristics --- Cache Performance

- Cache Performance
  - Differences are minor (2-way, 4-way, 8-way associative caches)
  - Miss rates are different on each HPA/PPA applications
  - Miss rates decrease with increasing cache size



(a) Instruction Cache Average



(b) Data Cache Average

Design Implications

- Computational Complexity
  - Estimate the computational power of CPU
- Instruction Set Design
  - Optimize the instruction sets
- I/O Requirements for Multi-Processor ASIC
  - Estimate the memory bandwidth

Example 1 --- Computation Complexity

- Requirements of RTR and DRR
  - Link bit rate 1.2Gb/s
  - Packet size is 576 bytes
  - So, computation requirement is
    - $M = (N_{RTR,\ 576} + N_{DRR,\ 576}) * R_{Lnk}$
      $= (0.2+0.5)$ instr/byte * 1.2G/8 bytes/sec
      $= 105$ MIPS
- Requirements of on-the-fly CAST encryption
  - Link bit rate 1.2Gb/s
  - Ignore header processing overhead
  - So, computation requirement is
    - $M = N_{CAST,¥} * R_{Lnk}$
      $= 104$ instr./byte * 1.2G/8 bytes/sec
      $= 15{,}600$ MIPS

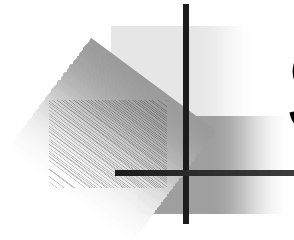
Example 2 --- Instruction Set Design

- Optimization instruction sets
- Special non-standard instructions

Instr. Pairs	Avg. occurrence	Max. occurrence
ADD-SUBCC	3.55%	11.6%
LD-SUBCC	3.03%	13.0%
LD-LD	2.36%	20.2%
ADD-LDUB	2.07%	4.84%
SLL-LD	2.05%	6.82%
STB-ADD	2.05%	4.65%
LD-ADD	1.85%	5.35%
ADD-Add	1.84%	5.85%

Example 3 --- I/O Requirements

- Multiple network processors
- Estimate the average memory bandwidth for application *a* and cache size *c*
  - $mbw_{a,c} = (I_{miss\ rate_{a,c}} + (D_{miss\ rate_{a,c}} * \%load_a) + \%store_a) * clock * line\ size$
- For *a* = *CAST* and *c* = *8k*, 400MHz ASIC, the memory bandwidth is
  - $mbw_{CAST,8k} = (0.0385+(0.0076 * 0.1985) + 0.0722) * 400 * 10^6 * 32\ bit/sec = 1.4\ Gbit/sec$



Summary

- CommBench supports evaluation and design of telecommunications network processors
- Various characteristics
  - Code and kernel size
  - Computational complexity
  - Instruction frequency
  - Cache performance
- Defined I/O and computational complexity