Scheduling Computations on a Software-Based Router

ECE 697J

November 19th, 2002

Processor Scheduling

- How is scheduling done on a workstation?
- What does the operating system do?
- What is the “cost” of scheduling?
- What can be guaranteed through scheduling?
- How is this different on a router?

Scheduling Problem

- Problem on routers: “Live-Lock”
  - Packets on interface cause interrupt
  - Interrupt processing hogs processor
  - No forwarding is done
  - All packets get dropped
- Scheduler should provide different levels of service
- How to split processing into components?
- How to assign proper ratio of processing time?

Scheduler I

- One process
- Steps of processing:
  - READ: read packet from input port
  - CLASSIFY: select output port
  - PROCESS: perform required packet processing
  - WRITE: write packet to output port
- Problems:
  - FIFO order, no service differentiation
  - If PROCESS takes too long packets get dropped

Scheduler II

- Separate traffic classes
  - Multiple queues
- Process queues individually
- Schedule among queues for output port
- Process assignment:
  - I: READ + CLASSIFY + ENQUEUE
  - O: SELECT + DEQUEUE + WRITE
  - F: FORWARDING

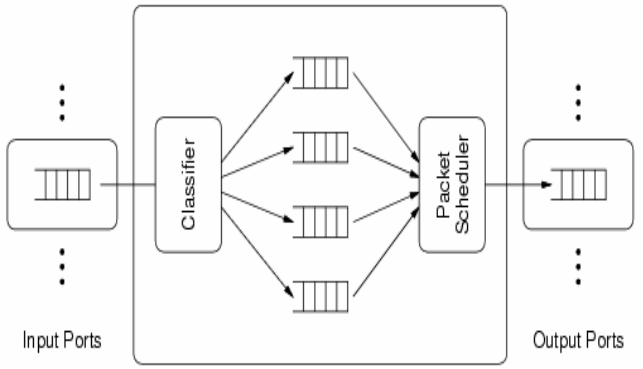


Figure 1: Supporting Differentiated Service

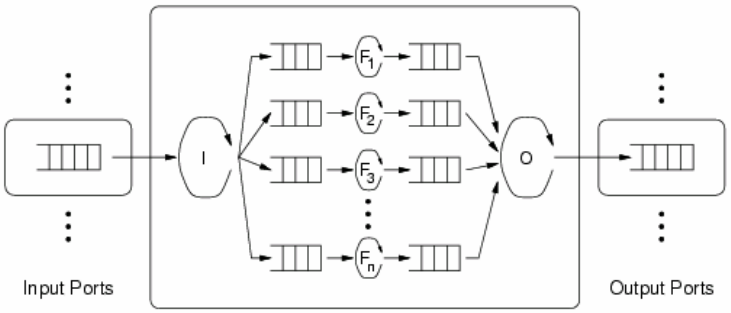


Figure 2: Supporting Differentiated Service and Variable Processing

Forwarding Function

- Forwarding in individual process
- Why not part of I or O?
  - Could take arbitrarily long
  - I could be blocked
  - O might underflow link
- How many forwarding processes?
  - Per packet? Too many
  - One? No separation of flows
  - Per flow!

Forwarding Function	Per-Packet Cost (μs)
IP Fast Path	0.3
General IP	3.0
Transparent Proxy	10.7
Classical Proxy	12.8
Active Protocol	37.3

Table 1: Costs of various forwarding functions, measured in microseconds, on a 450 MHz Pentium II. These times are independent of the costs of the input and output processes.

Forwarding Performance

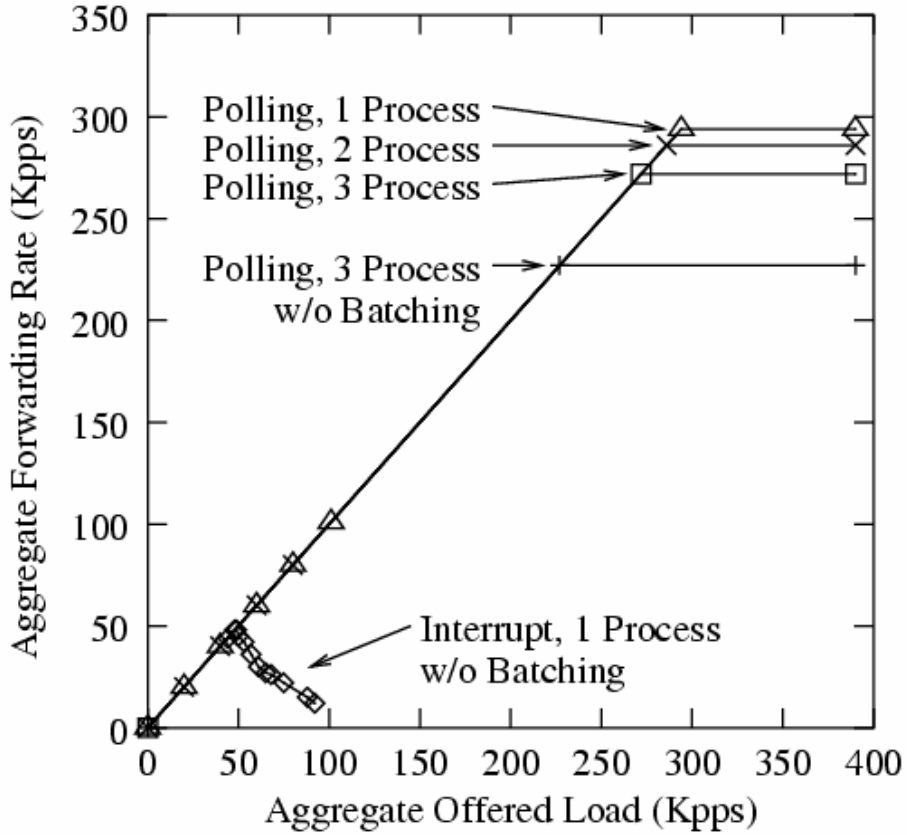


Figure 4: Impact of interrupt handling and context switching on forwarding rate.

Scheduling Processes

- Proportional Share (PS) scheduler:
  - Each processor reserves rate
  - Unused capacity is shared fairly without charge
  - Idle processes cannot save credits
  - Guaranteed rates provide isolation
- How should share be allocated to I,Fs,O?
- Important: “balance”
  - Too much to I causes overflow
  - Too much to O causes underflow
  - Live-lock is symptom of poor balance

Share Allocation

- Share allocation impacts guarantees:

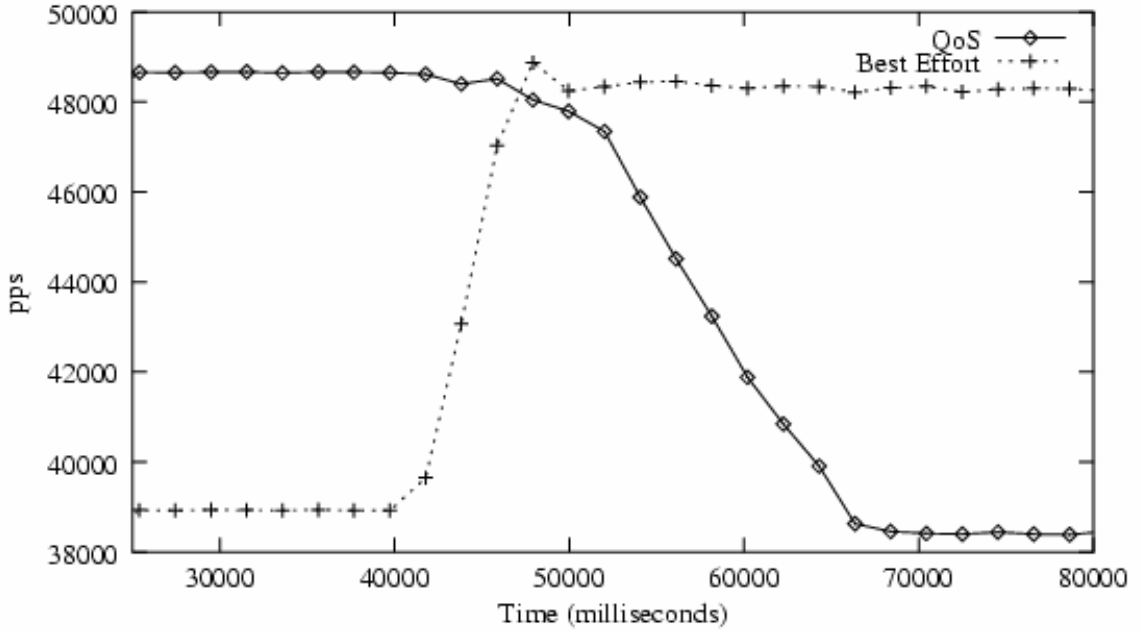


Figure 5: QoS experiencing drops under heavy load

- Conservative share to input process is preferable

Context Switching

- Cost of context switching is expensive (10 µs)
- How to reduce overhead?

Batching

- Allow processing of multiple packets before context switch
- Limit:
  - Number of packets
  - Amount of time spent
- Problem:
  - Coarser schedule
- Batching throttle:
  - Adjust batching granularity dynamically

Simple Batching

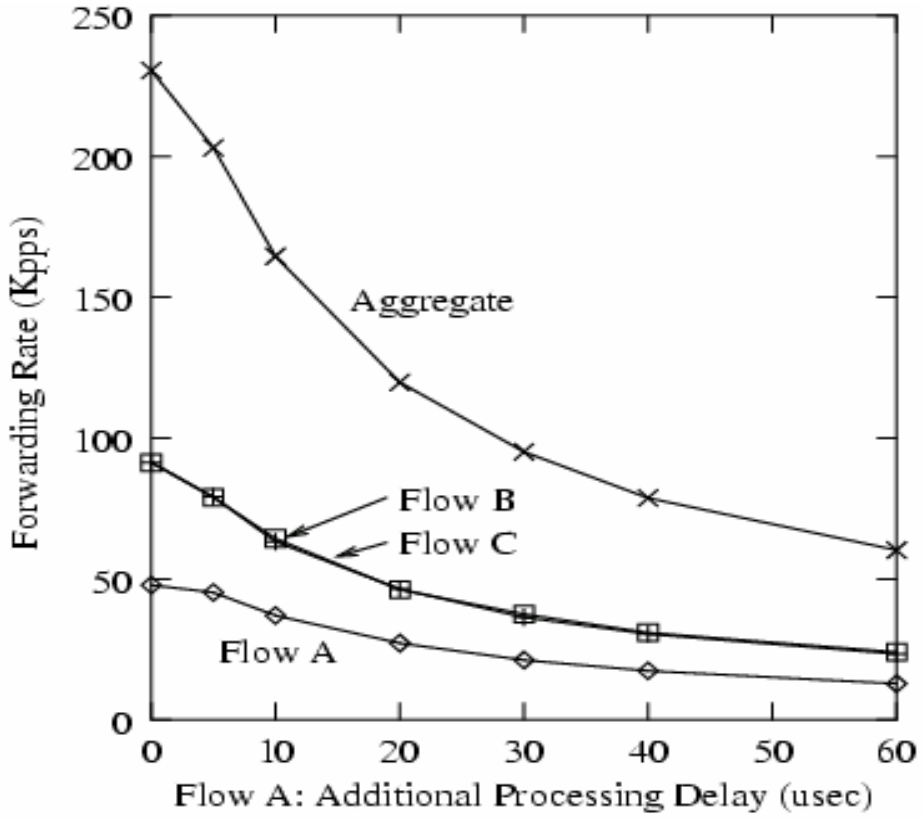


Figure 6: Detrimental Effects of Simple Batching as Processing Costs Increase.

No Batching

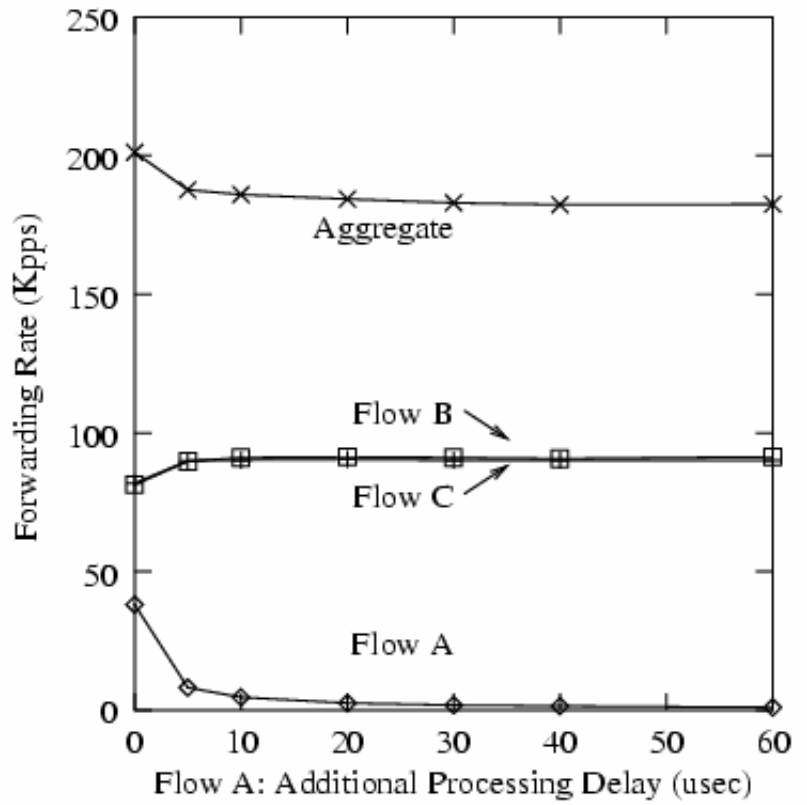


Figure 7: No Batching

Batching with Time Limit

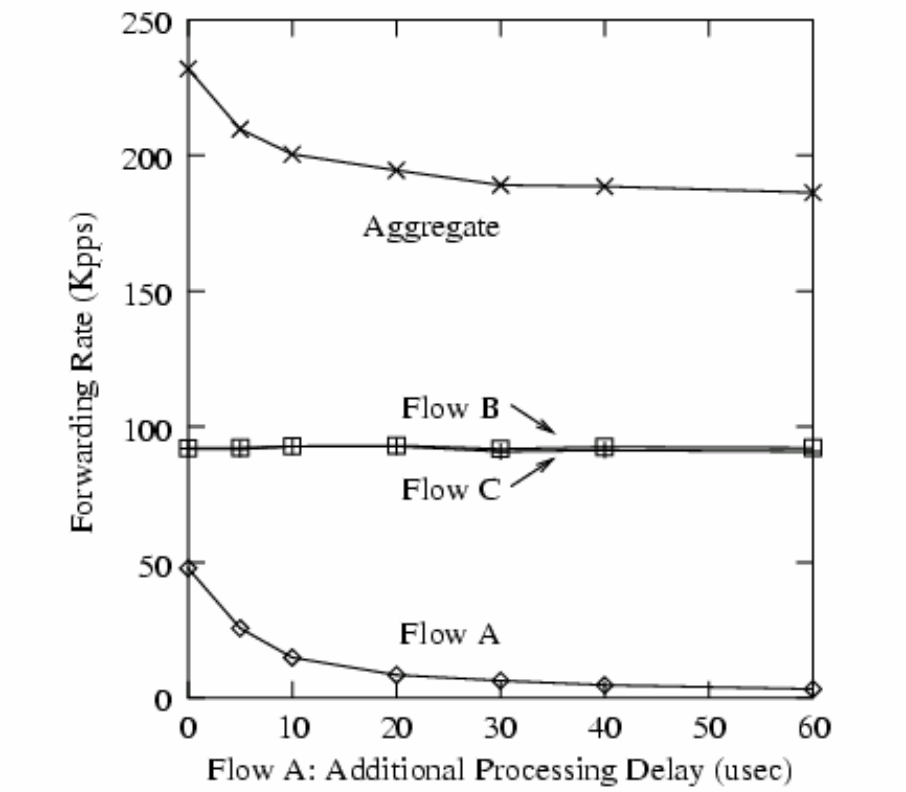


Figure 8: Batching with a 30 μs limit.

Batching Throttle

- Each flow can be scheduled according to its needs
  - Granularity G
  - Timeout D
  - Processing cost
- Context switch every G time units
- Flow becomes eligible once it's "efficient" (depends on C_sw)
- IF D time units have passed, process is also eligible

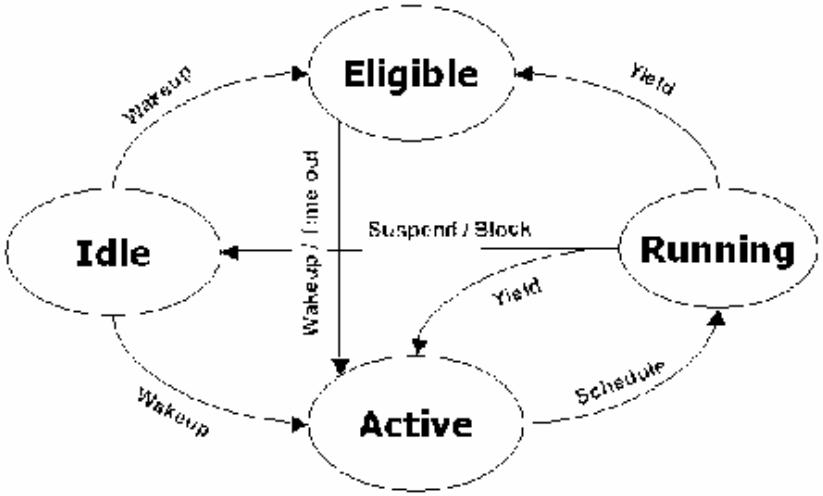


Figure 9: State Transition

Efficiency

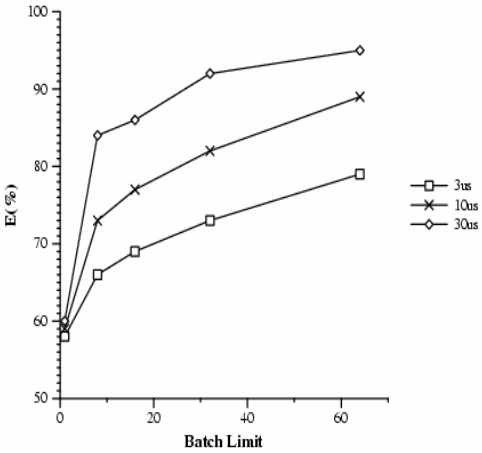


Figure 10: Efficiency Index for simple batching.

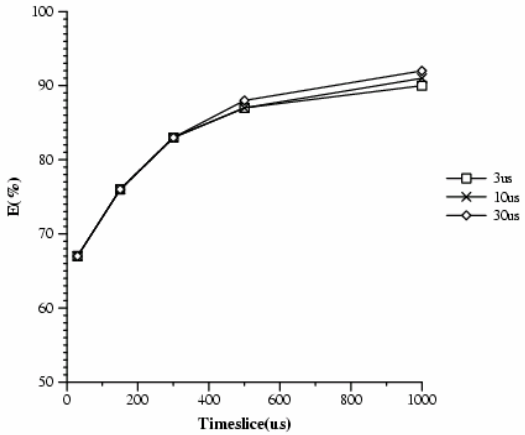


Figure 13: Efficiency Index for timeslice.

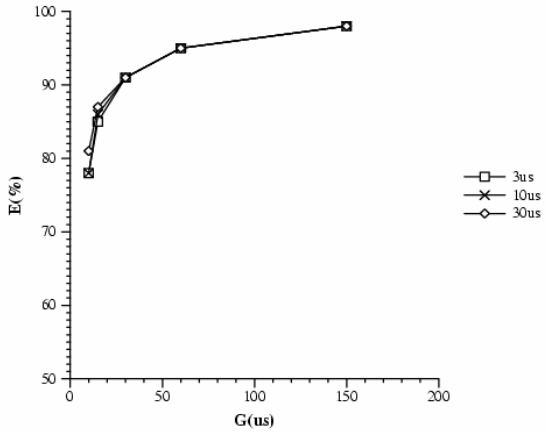
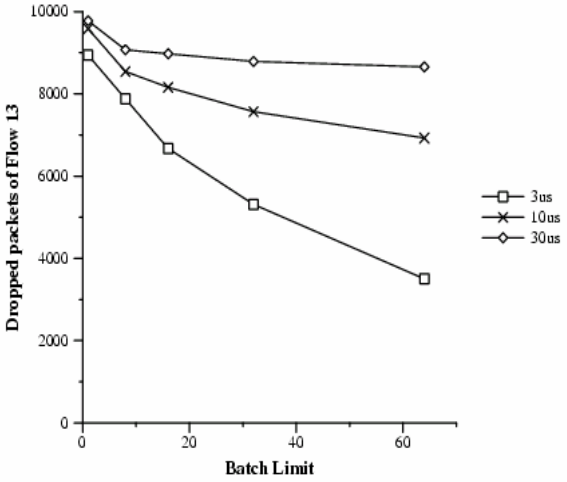


Figure 16: Efficiency Index for batching throttle.

Dropped Packets



gure 11: Flow 13's packets dropped for s
tching.

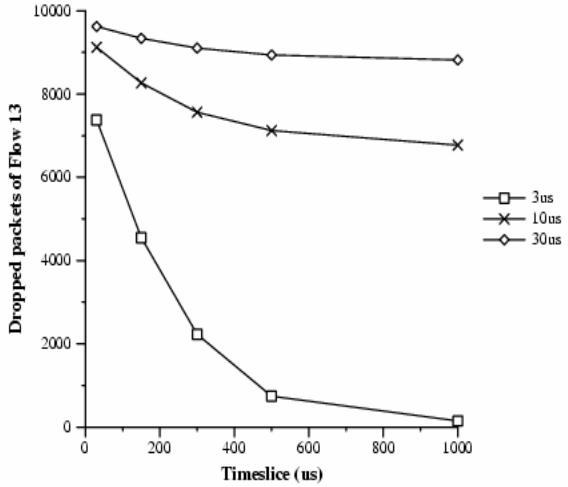


Figure 14: Flow 13's packets dropped when
timeslice.

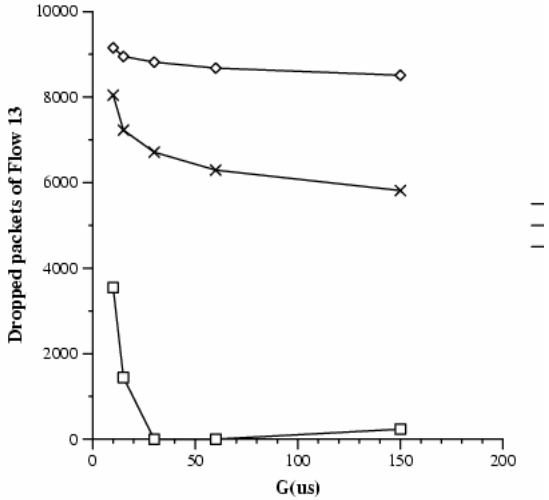


Figure 17: Flow 13's packets dropped w
batching throttle.

Queue Length (Delay)

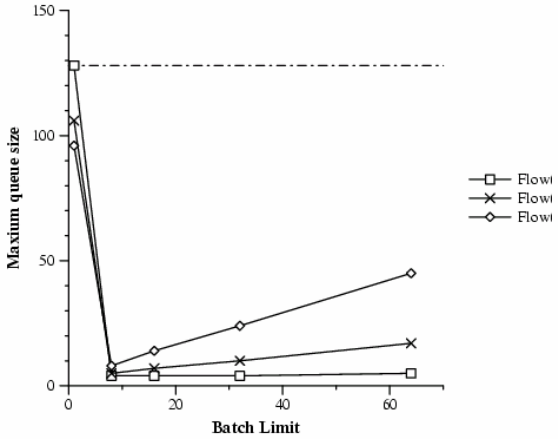


Figure 12: Maximum queue length seen by F

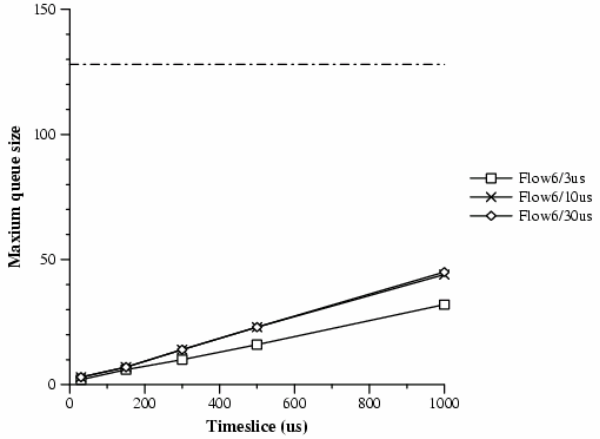


Figure 15: Maximum queue length seen by Flow 6 with timeslice.

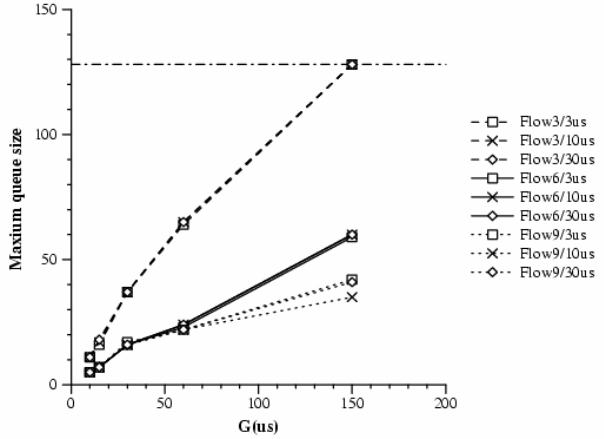


Figure 18: Maximum queue length seen by Flow 3, 6, 9 when using the batching throttle.

Summary

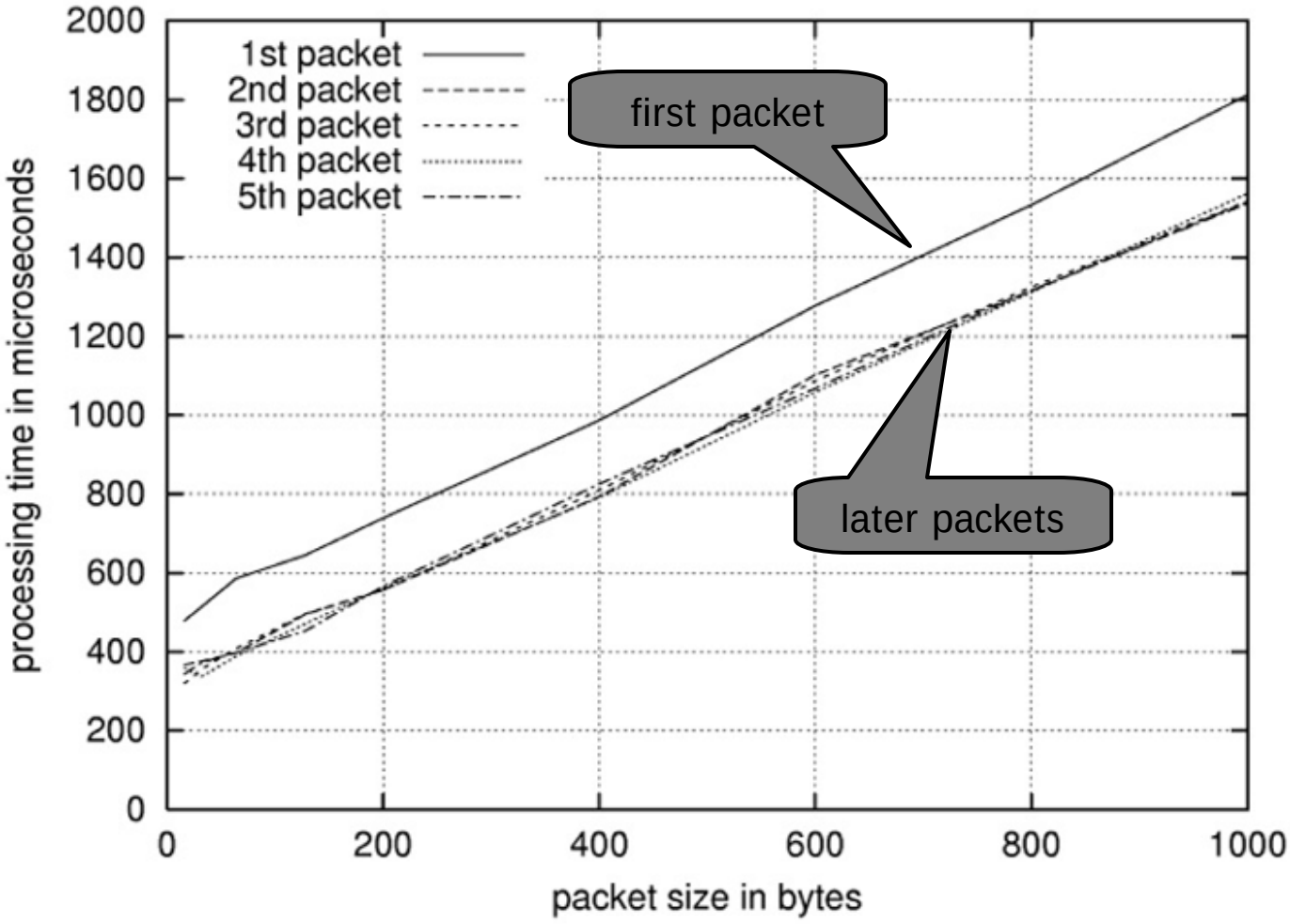
- Batching Throttle allows custom configuration of scheduler
  - Done by local policy
- Benefits:
  - Stable delay bound and throughput
  - Can be adapted to individual flow needs

Task Scheduling Problem

- Assignment of **packets to processors**
- **Novel problem**:
  - CPU scheduling unsuitable: assumes context switching
  - Bandwidth scheduling unsuitable: need exact processing time
- Affects **NP performance**:
  - Small caches can hold only one program
  - Change in application causes "**cold cache**"
  - Can create up to 25% overhead
- Processors can reuse instruction data
  - Scheduler can base decision on instruction locality

Performance Impact

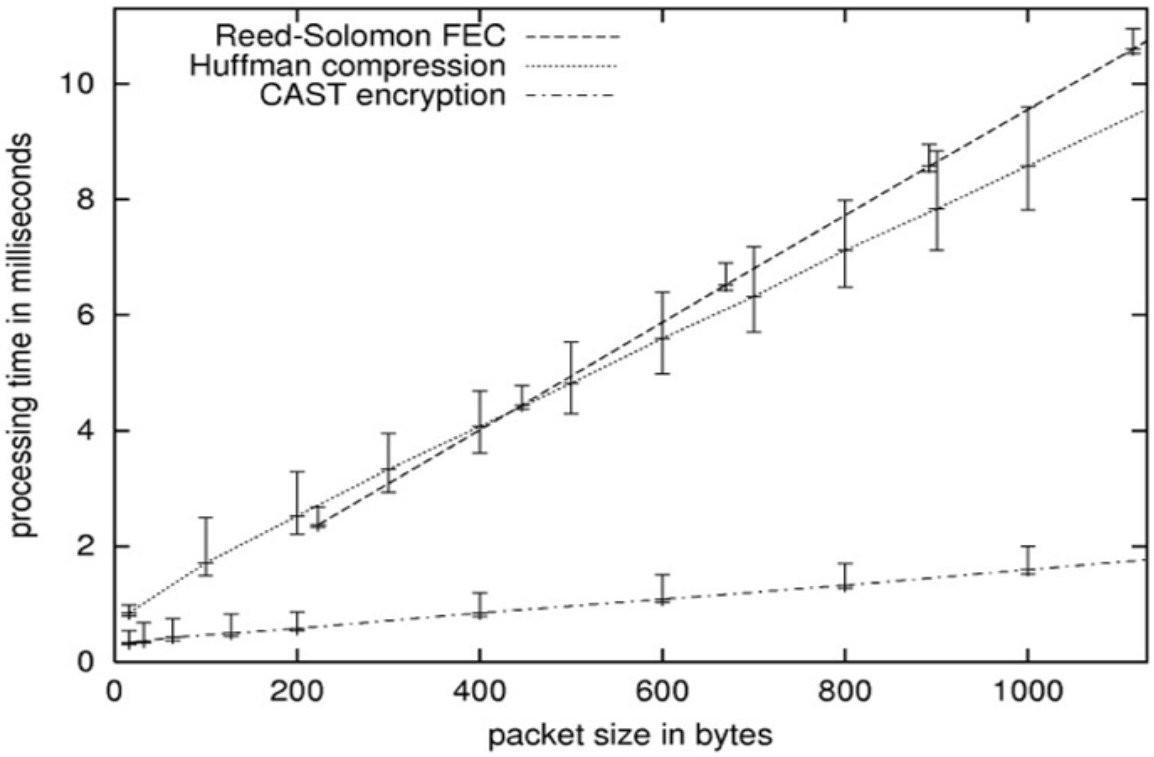
- **Measurement**:
  - SPC on WUGS
  - CAST app
- Constant **penalty** for first packet
- After 1st packet, cache is "warm"
- **Linear dependency** on packet size



Predictability

- Processing time can be **estimated**:

$$time = per\ packet\ cost + size \cdot per\ byte\ cost$$



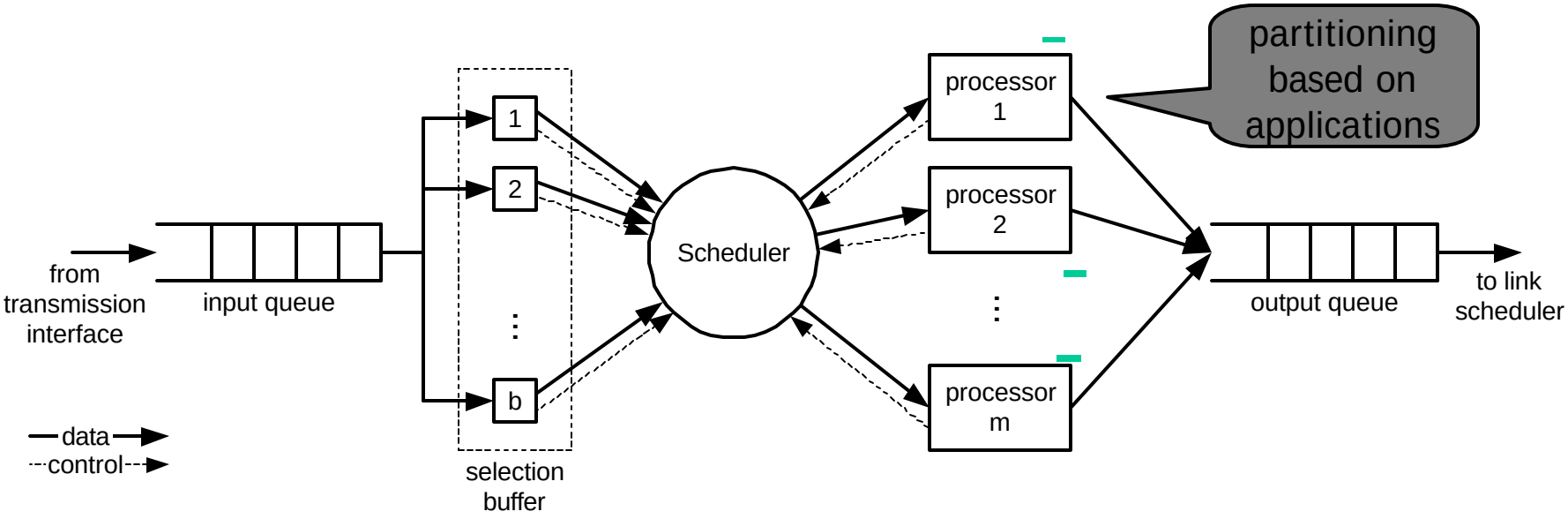
Application	per-packet cost in µs	per-byte cost in µs
Encryption	320	1.3
Compression	970	7.6
FEC coding	320	9.2

y-axis offset

slope

Locality-Aware Predictive Scheduling

- Partition processors into **application groups**
  - Reuse instruction cache in group
  - Partition sizes depends on application requirements
- When processor idle, pick packet with same application
- "Selection buffer" allows view of upcoming packets



Scheduling Summary

- LAP **approximates optimal** offline scheduling for large buffers
- LAP can achieve ~20% **higher throughput** than FIFO by avoiding cold cache penalties
- Complexity is **O(1)**
- Details: [Wolf, Franklin: "Locality-Aware Predictive Scheduling for Network Processors," IEEE ISPASS 2001.]

- Other scheduling algorithm: **Estimation-based Fair Queuing**
- Processing time estimation:
  - **Resource reservation**
  - **Fair sharing** of processing resources
  - **Delay bounds** for conforming flows
- Details: [Pappu, Wolf: "Scheduling Processing Resources in Programmable Routers," IEEE INFOCOM 2002.]

Next Lecture

- Programmable Logic (JK)
- Choose one NP for class on 12/05/02:
  - Motorola C-5E
  - EZ-Chip NP-1
  - Vitesse IQ2000
  - AMCC nP7250
  - Agere Payload Plus
  - Cisco Toaster 2
  - Lexra NetVortex PowerPlant
  - Terago proNP 5010