Wormhole IP over ATM

Manolis Katevenis e.a.

FORTH and University of Crete, Greece

http://archvlsi.ics.forth.gr/wormholeIP/

Work performed together with:

- Iakovos Mavroidis
- Georgios Sapountzis
- Thanos Papathanasiou
- Georgios Kalokerinos
- Georgios Glykopoulos
- Ioannis Mavroidis

and with the assistance of:

- Evangelos Markatos
- Vassilios Siris
- Fotis Kitsos
- Stelios Sartzetakis

How to speed up IP Routers?

- higher throughput
- lower latency
- quality of service (QoS)
- low cost

move from: general-purpose-computer routers

⇓

to: routers with hardware assistance

which means:

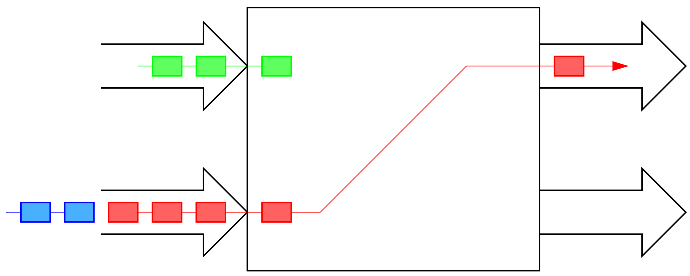
- Routing Table Lookup's in hardware
  - feasible: see Gupta e.a. '98, Moestedt e.a. '98, etc.
- Switching in hardware
  - done, but works on small, fixed-size quanta...

Variable-Size Packets over Fixed-Size Hardware?

known technology:

- in the 80's: *Wormhole Routing*

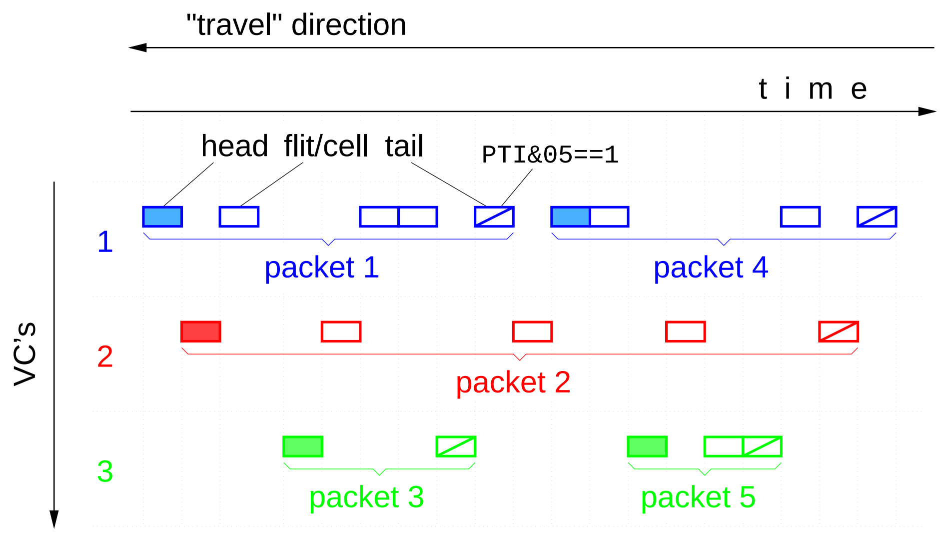
  most popular multiprocessor interconnection network
  - packets segmented into "flits"
  - flits forwarded using: *Virtual Cut-Through:*



- in the 90's: *ATM Hardware*
  - packets segmented into "cells"

Similarity between Wormhole Routing and ATM (AAL5)

Packets segmented into flits or cells and interleaved on multiple VC's



Outline

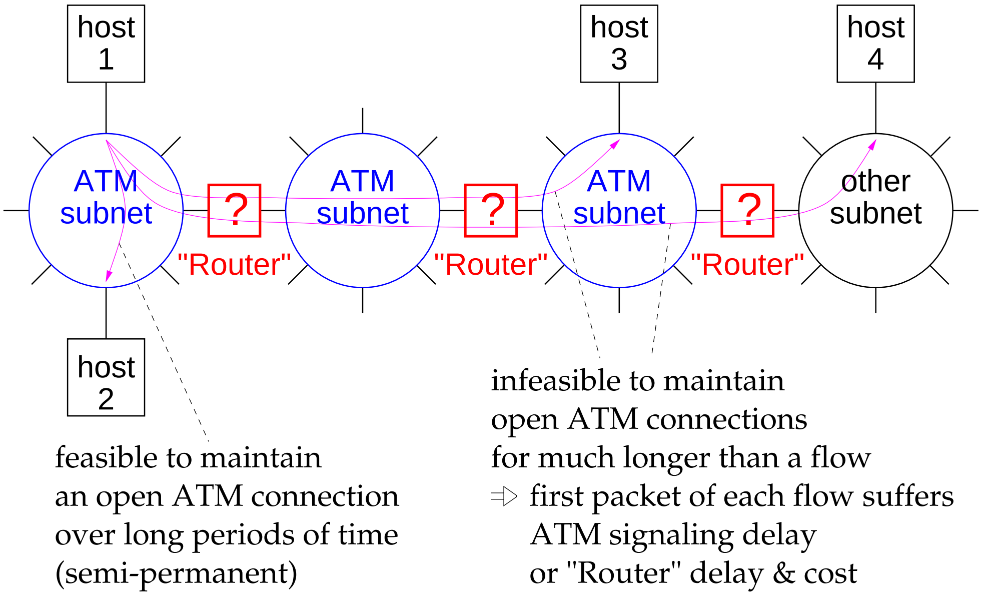
- *How it works*
  - build wormhole routers -like those of the 80's- for IP;

  or (this proposal):

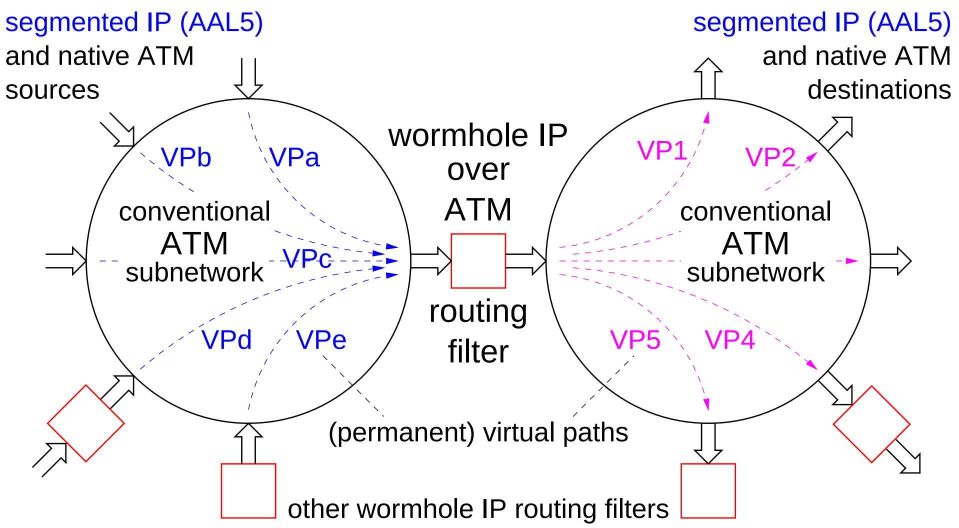
  ⇨ use existing ATM switches/subnetworks for forwarding,

  ⇨ design and build "wormhole routing filters", that map IP destination addresses into next-hop VP numbers
    - inexpensive
    - fast
      - (virtual cut-through)
      - (routing table lookup in hardware)
- *QoS*
- *Performance simulation*
- *Hardware prototype*

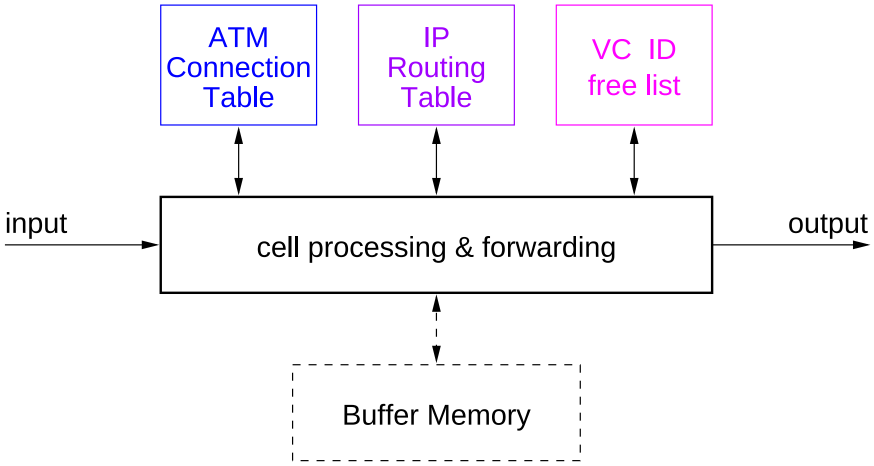
How to speed up IP traffic over ATM WAN at low cost?



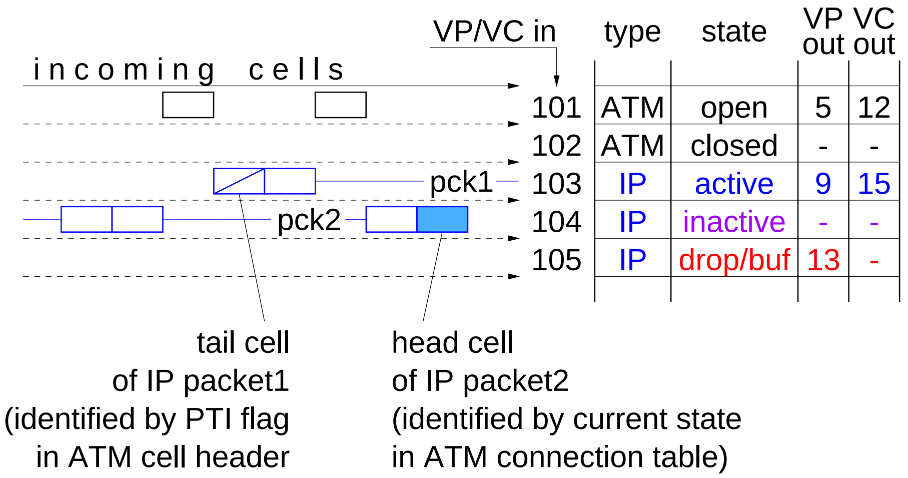
Wormhole IP Router as a Filter between two ATM Subnetworks



Functions in a Wormhole IP over ATM Routing Filter



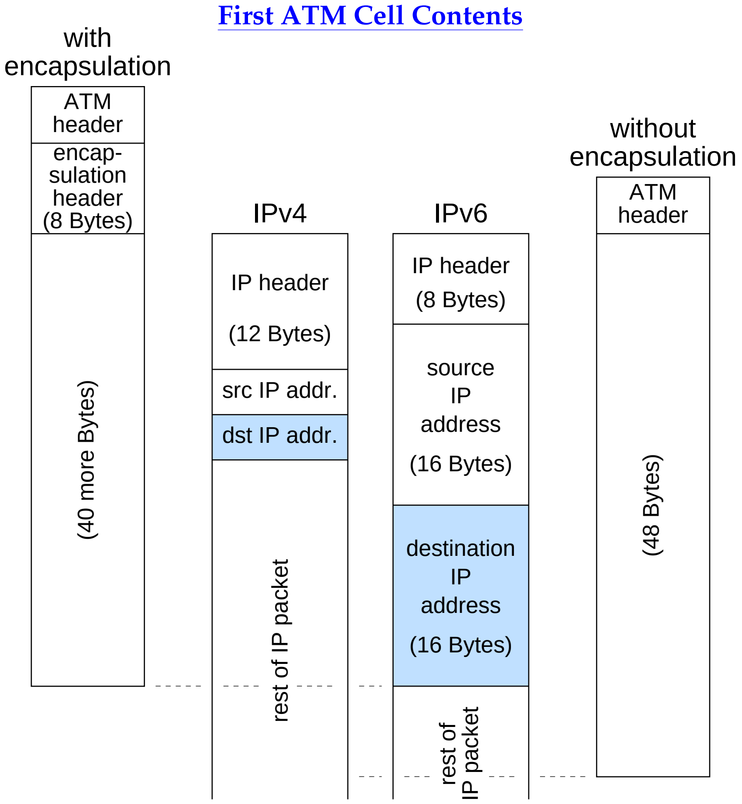
ATM Connection Table Contents

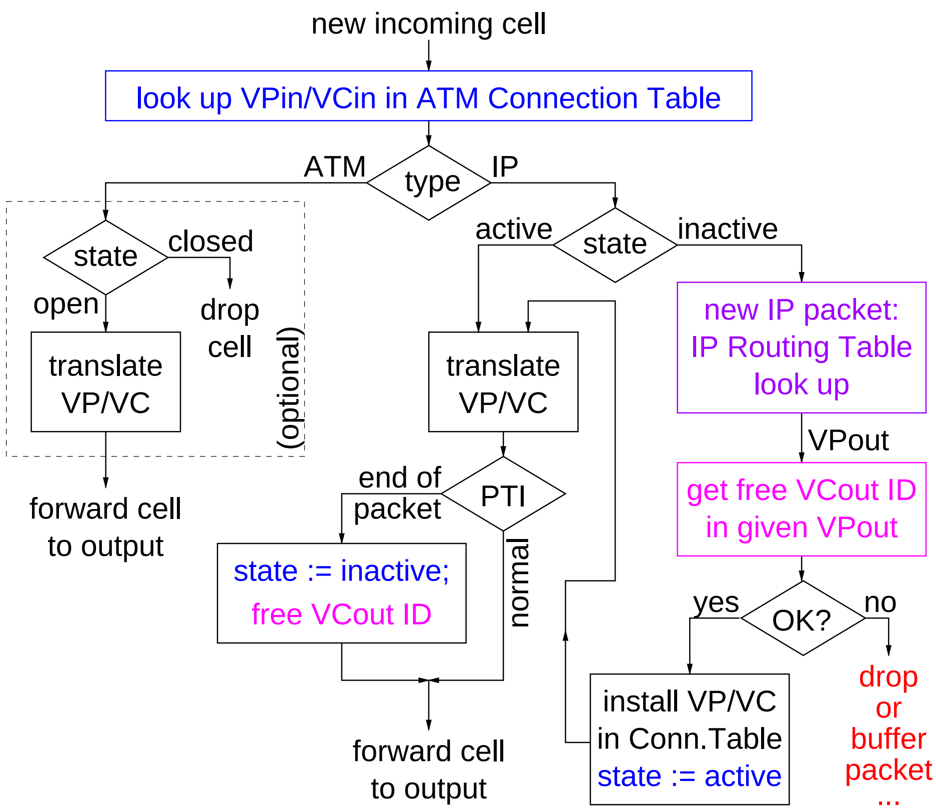


- connections (VP/VCin to VP/VCout) only last for one IP packet
- upstream VC (label) allocation, in hardware, w/o downstream ack

IP packet segmented according to AAL5:

First ATM Cell Contents





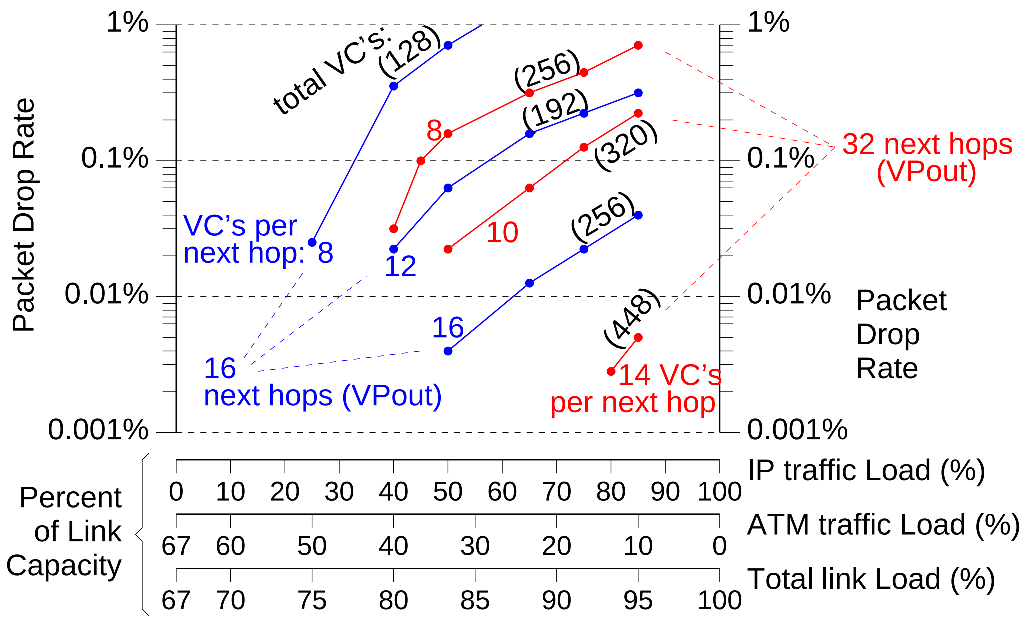
Quality of Service (QoS)

- ATM traffic maintains its QoS guarantees:
  - priority(ATM-type VP) > priority (IP-type VP) (ATM traffic subject to admission control & policing)
- IP traffic can benefit from QoS capabilities of ATM hardware:
  - multiple VP's / destination, @ different service class each
  - select VPout based on both:
    - destination address, and
    - flow service class
  - IP flow classifier in hardware:
    - in the wormhole routing filter itself, or
    - upstream of it: class(VPout) = class(VPin)

Bufferless Wormhole IP Routing Filter

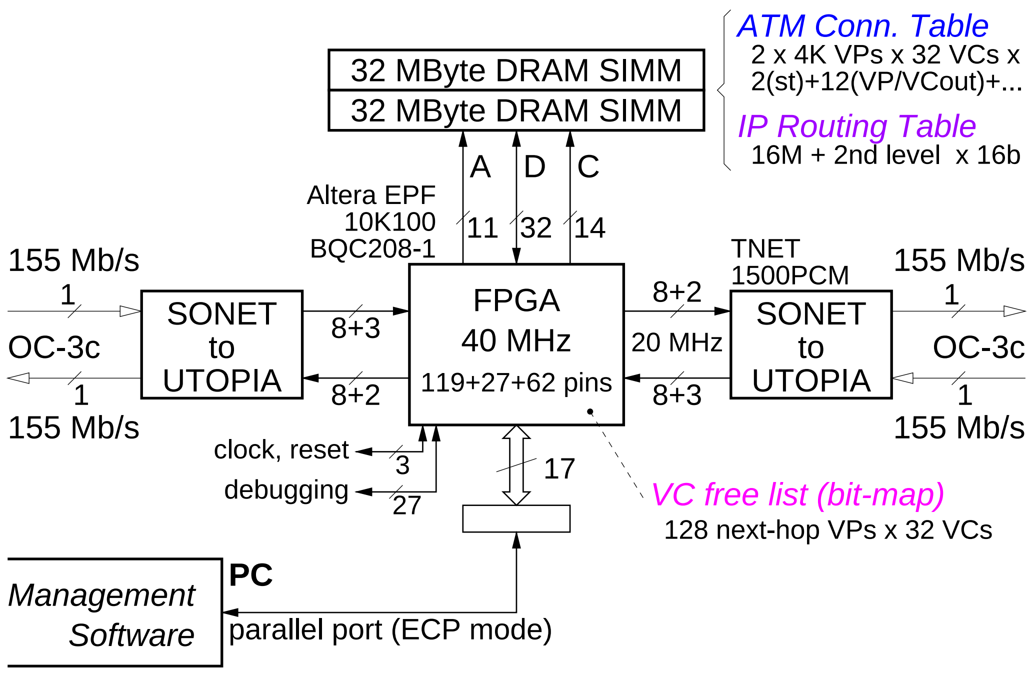
- *Bufferless:*
  - no cell (packet) buffer - simple & inexpensive
  - immediately forward all cells - ''zero'' delay
  - when all VC's are busy on desired next-hop VP: just drop the packet!
- How many VC's per next-hop VP, for the packet-drop rate to be very low?
  - simulation, using trace of real IP backbone traffic
  - the cells of each packet were spread in time to simulate the effect of the ATM subnetwork(s)

Bufferless Routing Filter Performance Simulation

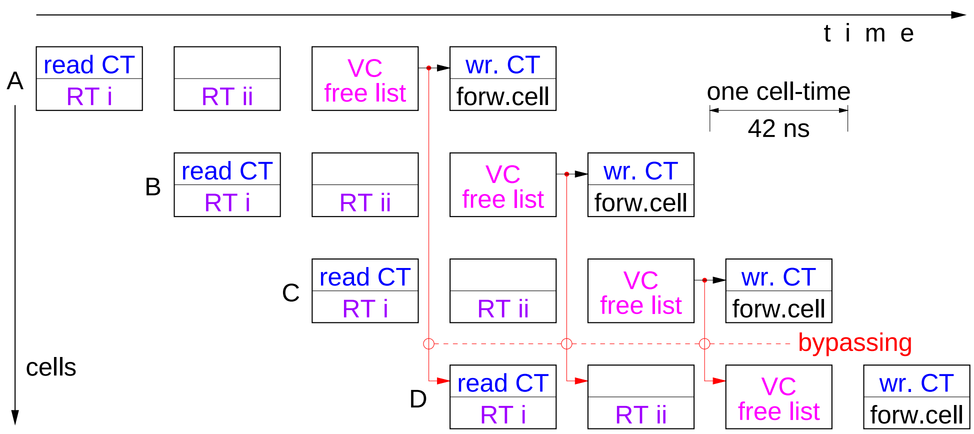


"FIXWEST" Internet Backbone Router traffic (Jan.97); 300,000 cell times simulated
average intra-packet cell-to-cell spreading = 50 cell times

Block Diagram of Hardware Prototype:
2x155 Mbps Bufferless Wormhole IP Routing Filter



Pipeline Example for Wormhole IP over ATM Operation at 10 Gb/s



Conclusions

- *Wormhole Routing:* well known architecture for
  low-latency routing of variable-size packets,
  over low-cost hardware, using fixed-size quanta
- *Wormhole IP over ATM:* low-cost method to turn existing
  ATM networks into low-latency gigabit IP routers
- IP and native-ATM traffic share the same link capacity
- ATM enjoys full QoS; IP can benefit from ATM's QoS capabilities
- microsecond routing latency for all IP packets
- a few tens of (hardware-managed) VC's per VPout suffice
- few chips for gigabit rates; 10 Gb/s rate feasible, today