COMMUNICATIONS

Packet Scheduling in Next-Generation Multiterabit Networks

Itamar Elhanany, Michael Kahane, and Dan Sadot
Ben-Gurion University

The infrastructure required to govern Internet traffic volume, which doubles every six months, consists of two complementary elements: fast point-to-point links and high-capacity switches and routers. Dense wavelength division multiplexing (DWDM) technology, which permits transmission of several wavelengths over the same optical media, will enable optical point-to-point links to achieve an estimated 10 terabits per second by 2008. However, the rapid growth of Internet traffic coupled with the availability of fast optical links threatens to cause a bottleneck at the switches and routers.

SWITCH FABRICS

Switches and routers consist of line cards and a switch fabric. A network processor that interfaces to the physical data links, decodes incoming traffic, and applies traffic shaping, filtering, and other policy functions resides on each line card, which is associated with a port.

Actual data transmission across ports occurs on the switch fabric, which includes a crosspoint, a scheduler, and buffers. The crosspoint is a configurable interconnecting element that dynamically establishes links between input and output ports. The scheduler configures the crosspoint, allows buffered data to traverse, and repeatedly reconfigures the crosspoint for successive transmissions.

Multiterabit networks will require innovative queuing strategies and high-performance scheduling algorithms to meet the future packet-scheduling challenge.

Optical versus electrical

A major debate exists on whether next-generation multiterabit routing scenarios should use optical or electrical switch fabrics. At first glance, all-optical switches are the straightforward solution because they enable high-speed multiterabit aggregation and concentration. However, emerging router functions, such as quality-of-service (QoS) provisioning, require that packets be buffered—typically at the network processors—until the scheduler grants transmission. Because dynamic optical buffering is impractical, delayed transmissions currently dictate using opto-electric and electro-optical conversions.

Switching nodes, which entail simplified buffer and transmission management, more efficiently process network protocols deploying fixed packet sizes such as asynchronous transfer mode (ATM). The network processor typically segments variable-length packets, such as IP packets, into smaller, fixed-size data units that traverse the switch core and later reassembles them at the output ports for transmittal in their original format.

The length of data units passing through the switch core directly affects switch architecture and performance. Larger data units relax the timing requirements imposed on the switching and scheduling mechanisms, while smaller-sized units offer finer switching granularity. Designs commonly use 64-byte data units as a trade-off.

QoS classes

Accompanying the rapid pace of Internet traffic is the increasing popularity of multimedia applications sensitive to mean packet delay and jitter. To provide differentiated services, developers categorize incoming packets into QoS classes. For each QoS class, the router must not exceed the acclaimed latency and jitter requirements.

For routers to operate under heavy traffic loads while supporting QoS, designers must implement smarter scheduling schemes, a difficult task given that routers usually configure the crosspoint and transmit data on a per-data-unit basis. The scheduling algorithm must make a new configuration decision with each incoming data unit. In an ATM switch, where the data unit is a 53-byte cell, the algorithm must issue a scheduling decision every 168 ns at 2.5-Gbit-per-second line rates. As 10-Gbit-per-second port rates become standard for high-end routers, the decision time reduces fourfold, to a mere 42 ns.

QUEUING STRATEGIES

The switch fabric core embodies the crosspoint element responsible for matching N input and output ports. Currently, routers incorporate electrical crosspoints but optical crosspoint solutions, such as those based on dynamic DWDM, show promise. Regardless of the underlying technology, the basic functionality of determining the crosspoint configuration and transmitting data remains the same.

Input queuing

We can generally categorize switch-queuing architectures as input- or output-queued. Network processors store arriving packets in input-queued switches in FIFO buffers that reside at the input—or ingress—port until the processor signals them to traverse the crosspoint.

A disadvantage of input queuing is that a packet at the front of a queue can prevent other packets from reaching potentially available destination—or egress—ports, a phenomenon called head-of-line (HOL) blocking. Consequently, the overall switching throughput degrades significantly: For uniformly distributed Bernoulli iid traffic flows, the maximum achievable throughput using input queuing is 58 percent of the switch core capacity.

Virtual output queuing (VOQ) entirely eliminates HOL blocking. As Figure 1 shows, every ingress port in VOQ maintains N separate queues, each associated with a different egress port. The network processor automatically classifies and stores packets upon arrival in a queue corresponding to their destination. VOQ thus assures that held-back packets do not block packets destined for available outputs. Assigning several prioritized queues instead of one queue to each egress port renders per-class QoS differentiation.

Output queuing

Output-queuing strategies directly transfer arriving packets to their designated egress ports. A contention resolution mechanism handles cases in which two or more ingress ports simultaneously request packet transmission to the same egress port.

A time-division-multiplexing (TDM) solution accelerates the switch fabric internal transmission rates by N with respect to the port bit rates. The growing prevalence of 10- and 40-Gbit-per-second rates, however, makes acceleration by N infeasible.

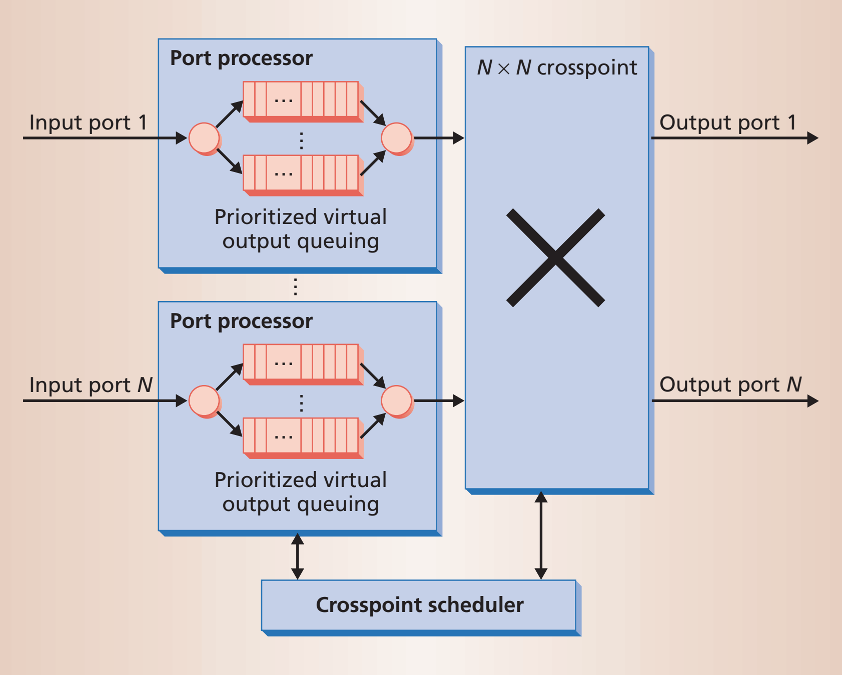

Figure 1. Router-switch architecture consisting of virtual output queuing buffering, crosspoint, and scheduling modules. The network processor automatically classifies packets arriving at the input ports to queues corresponding to their destination. The crosspoint scheduler receives transmission requests from the various queues and determines which queue is granted transmission at each time slot. Packets accordingly traverse the crosspoint to their designated output ports and depart the switch.

Trading space for time by deploying space-division multiplexing (SDM) techniques requires a dedicated path within the switch fabric for each input-output pair. However, SDM implies having $O(N^2)$ internal paths, only N of which can be used at any given time because at most, N input ports simultaneously transmit to N output ports. This requirement renders SDM impractical for large port densities such as $N = 64$.

SCHEDULING APPROACHES

The main challenge of packet scheduling is designing fast yet clever algorithms to determine input-output matches that, at any given time:

- maximize switch throughput utilization by matching as many input-output pairs as possible,
- minimize the mean packet delay as well as jitter,
- minimize packet loss resulting from buffer overflow, and
- support strict QoS requirements in accordance with diverse data classes.

Intuitively, these objectives appear contradictory. Temporarily maximizing input-output matches, for example, may not result in optimal bandwidth allocation in terms of QoS, and vice versa. Scheduling is clearly a delicate task of assigning ingress ports to egress ports while optimizing several performance parameters.

Moreover, as port density and bit rates increase, the scheduling task becomes increasingly complex because more decisions must be made during shorter time frames. Advanced scheduling schemes exploit concurrency and distributed computation to offer a faster, more efficient decision process.

PIM and RRM

Commonly deployed scheduling algorithms derive from parallel iterative matching (PIM), an early discipline developed by the Digital Equipment Corp. for a 16-port, 1-Gbit-per-second switch (http://www.cs.washington.edu/homes/tom/atm.html). PIM and its popular derivatives use randomness to avoid starvation and maximize the matching process. Unmatched inputs and outputs contend during each time slot in a three-step process.

- *Request.* All unmatched inputs send requests to every output to which they have packets to send.
- *Grant.* Each output randomly selects one of its requesting inputs.
- *Accept.* Each input randomly selects a single output from among those outputs that granted it.

By eliminating the matched pairs in each iteration, PIM assures convergence to a maximal match in $O(\log N)$ iterations. It also ensures that all requests are eventually granted. However, PIM has significant drawbacks, principally large queuing latencies in the presence of traffic loads exceeding 60 percent of the maximal switch capacity (calculated as number of ports multiplied by a port data rate). Moreover, PIM's inability to provide prioritized QoS and its requirement for $O(N^2)$ connectivity make it impractical for modern switching cores.

Developers designed the round-robin matching (RRM) algorithm to overcome PIM's disadvantages in terms of both fairness and complexity. Instead of arbitrating randomly, RRM makes selections based on a prescribed rotating priority discipline. Two pointers update after every "grant" and "accept." RRM is a minor improvement over PIM, but its overall performance remains poor under nonuniformly distributed traffic loads because of pointer synchronization.

iSLIP

iSLIP, the widely implemented iterative algorithm developed by Stanford University's Nick McKeown (http://www-ee.stanford.edu/~nickm), is a popular descendant of PIM and RRM that consists of the following steps:

- *Request.* All unmatched inputs send requests to every output to which they have packets to send.
- *Grant.* Each output selects a requesting input that coincides with a predefined priority sequence. A pointer indicates the current location of the highest-priority elements and, if accepted, increments (modulo N) to one beyond the granted input.
- *Accept.* Each input selects one granting output according to a predefined priority order. A unique pointer indicates the position of the highest-priority element and increments (modulo N) to one location beyond the accepted output.

Advanced scheduling schemes exploit concurrency and distributed computation to offer a faster, more efficient decision process.

Instead of updating after every grant, the outer pointer updates only if an input accepts the grant.

iSLIP significantly reduces pointer synchronization and accordingly increases throughput with a lower average packet delay. The algorithm does, however, suffer from degraded performance in the presence of nonuniform and bursty traffic flows, lack of inherent QoS support, and limited scalability with respect to high port densities. Despite its weaknesses, iSLIP's low implementation complexity promotes its extensive deployment side by side with various crosspoint switches.

Central prioritized scheduling

The independent pointers of request-grant-accept-based algorithms such as PIM and iSLIP tend to synchronize, degrading overall performance. An alternative approach, based on a centralized scheduling module, replaces the pointer mechanisms with a priority-generating function at each ingress port. During each time slot, the central module gathers prioritized requests from the input ports and produces matching decisions using global contention logic. Although more difficult to implement, this scheme yields lower packet delay at the expense of longer switching intervals. It also introduces lower connectivity, global contention resolution, and inherent QoS support.

Multiterabit packet-switched networks will require high-performance scheduling algorithms and architectures. With port densities and data rates growing at an unprecedented rate, future prioritized scheduling schemes will be necessary to pragmatically scale toward multiterabit capacities. Further, support of strict QoS requirements for the diverse traffic loads characterizing emerging multimedia Internet traffic will increase. Continuous improvements in VLSI and optical technologies will stimulate innovative solutions to the intricate packet-scheduling task. ✶

***Itamar Elhanany** is a PhD student in the Computer and Electrical Engineering Department at Ben-Gurion University. Contact him at itamar@ieee.org.*

***Michael Kahane** is an MSc student in the Computer and Electrical Engineering Department at Ben-Gurion University. Contact him at michaelk@bgumail.bgu.ac.il.*

***Dan Sadot** is professor and head of the Communications Systems Engineering Department at Ben-Gurion University. Contact him at sadot@ee.bgu.ac.il.*

Editor: Upkar Varshney, Department of CIS, Georgia State University, Atlanta, GA 30002-4015; voice +1 404 463 9139; fax +1 404 651 3842; uvarshney@gsu.edu