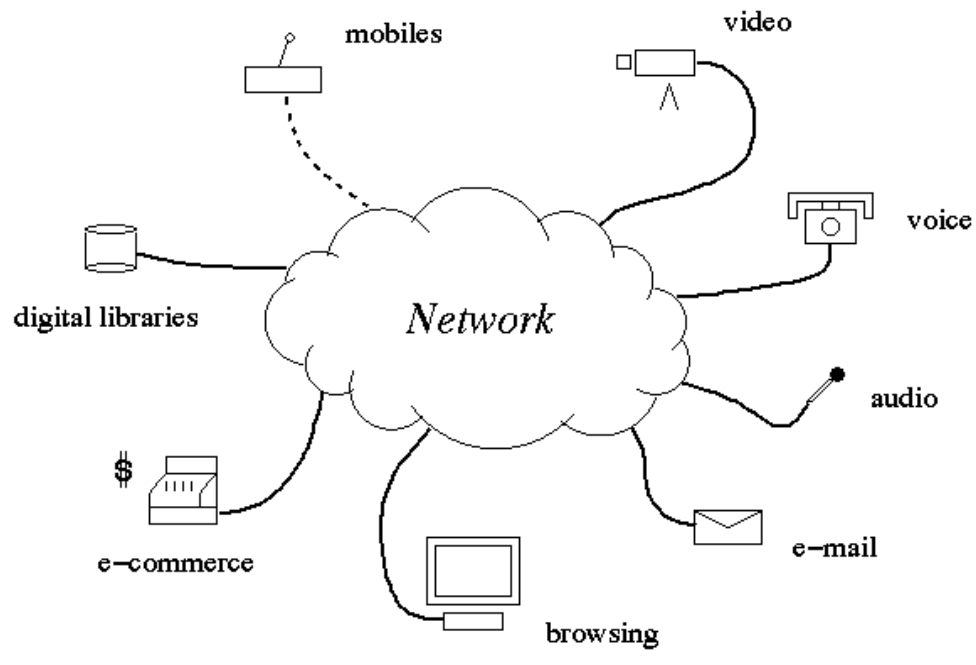
Molding Best-effort Bandwidth into Graded Services: The SBS Approach

Kihong Park
Network Systems Lab
Dept. of Computer Sciences
Purdue University
park@cs.purdue.edu

Joint: M. Sitharam, S. Chen, J. Cruz, H. Ren, T. Tuan

Motivation

- Network environment



Motivation (cont.)

- Diverse QoS requirements
- Shared network resources

Goal Facilitate QoS-sensitive sharing of network resources

Technical Challenges

- Heterogeneous QoS requirements
- Scalability
  - Aggregate flow (vs. per flow) control
  - Implicit admission control
- Efficiency
- User behavior: selfishness
- Implementability

SBS Approach

SBS: Stratified Best-effort Service

- Complement guaranteed services
  - "Small" population of stringent QoS applications
  - "Large" population of all other QoS-sensitive apps
  - Shares common substrate of GPS routers
  - Amortize cost of overprovisioning

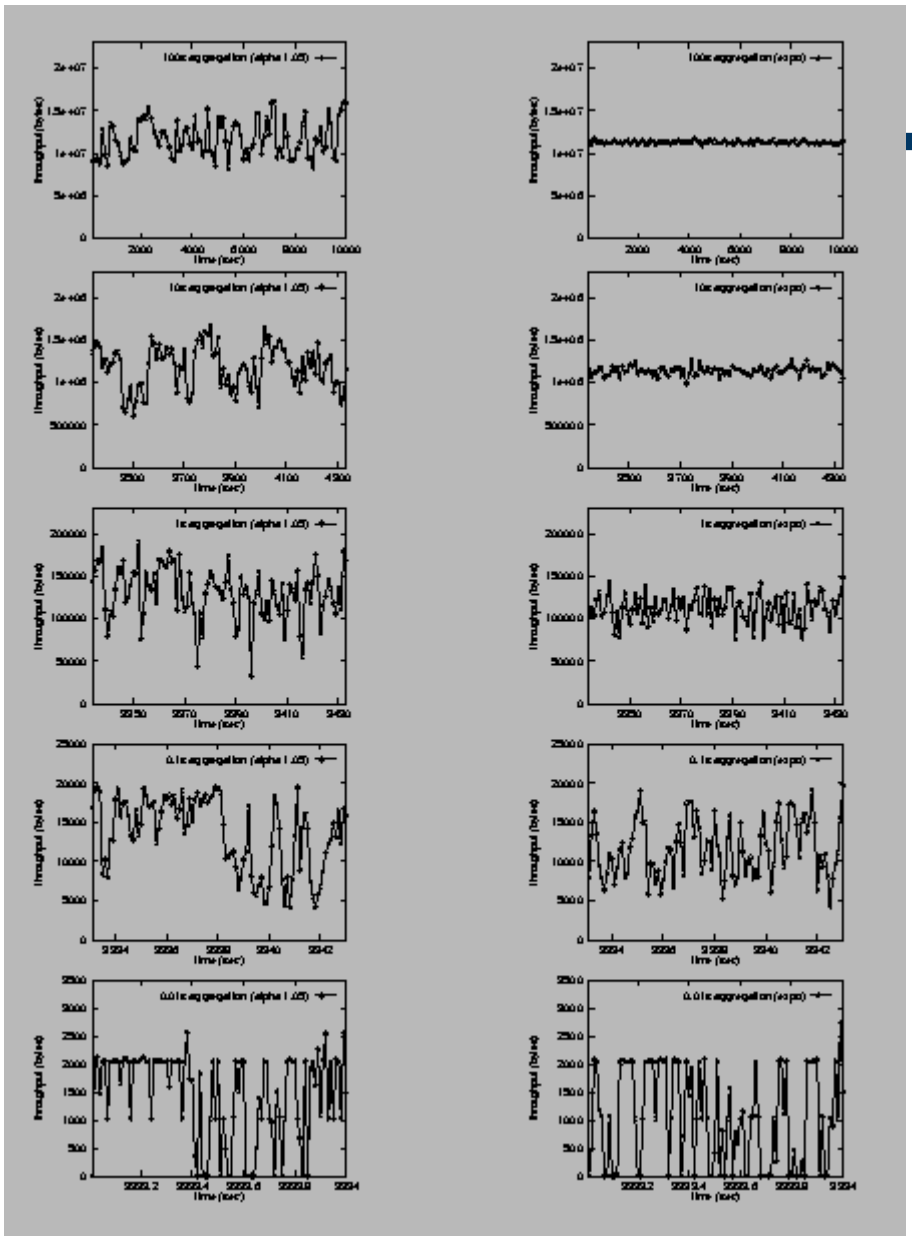
SBS Approach (cont.)

- Principle of noncooperative self-organization
  - Laisser faire fairness & efficiency
  - “Computational” invisible hand
- Shield resources from direct user control
  - Noncooperative network substrate: selfishness emulation
  - Well-defined, simple user/network interface
- Predictable but soft QoS
  - Weak protection
  - User interaction
- Scalar QoS control
  - WAN implementability

State-of-the-Art: Guaranteed Service

- Resource Reservation & Admission Control
  - On-demand leasing of bandwidth
  - Strong protection
  - QoS guarantees: deterministic/statistic
  - per flow/aggregate flow control
  - GPS routers
  - IETF Int-Serv (e.g., RSVP)

- Problems
  - Inefficient resource utilization: overprovisioning
  - Scalability



State-of-the-Art: Diff-Serv

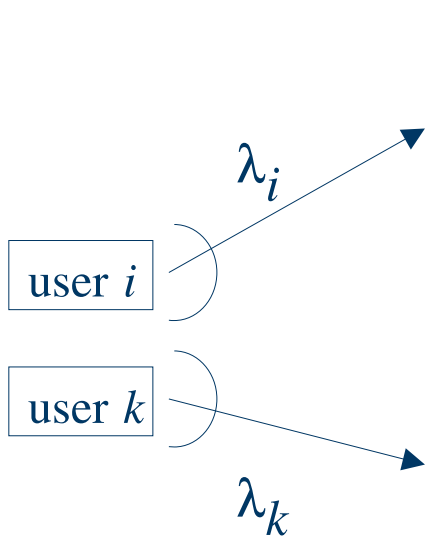
- IETF Diff-Serv
  - Assured service (Clark)
    - → IN/OUT marking with RIO routers
  - Expedited service (Jacobson)
    - → leaky-bucket with priority packet scheduling

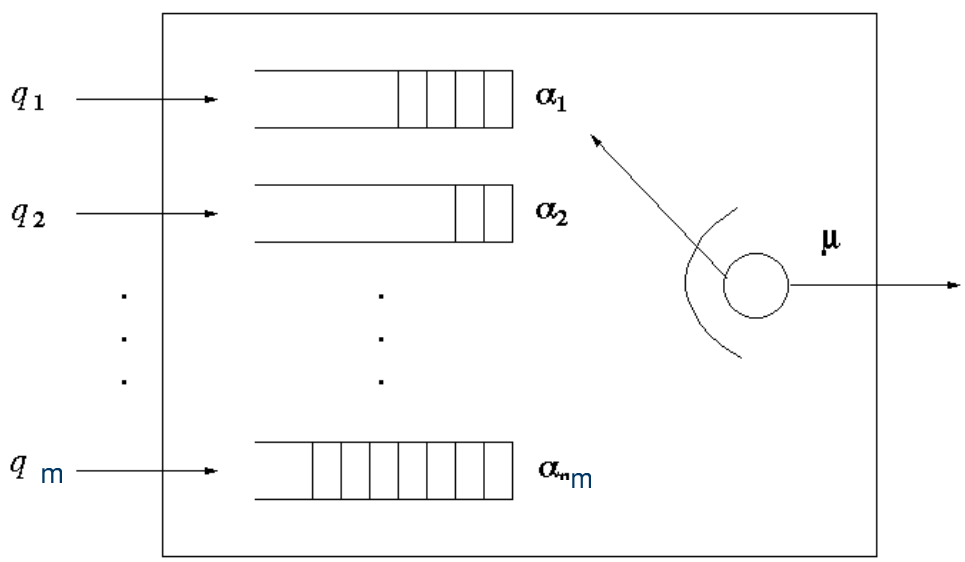
Central Theme Scalability through aggregate flow control

- Problems
  - "Differentiated" service ≠ QoS-sensitive service
  - Admission control: per flow contract computation

Single-switch System

- Multi-class QoS problem





GPS switch

Single-switch System (cont.)

- n users, m service classes
- $\partial c_j / \partial q_j \geq 0$; c_j: QoS indicator
- $\Lambda_i = (\lambda_{i1}, \lambda_{i2}, \ldots, \lambda_{im})$
- $\lambda_{ij} \in \{0, \lambda_i\}$; unsplittable
- $\Lambda = (\Lambda_1, \Lambda_2, \ldots, \Lambda_n)$; configuration

$\rightarrow$ QoS is induced via interaction

$\rightarrow$ leads to game

Single-switch System (cont.)

- Service class-QoS association table

Service Class	Measured QoS	Traffic Volume	Price
1	$\mathbf{c_1}$	q_1	p_1
2	$\mathbf{c_2}$	q_2	p_2
⋮	⋮	⋮	⋮
m	$\mathbf{c_m}$	q_m	p_m

$$\therefore \quad C_a \leq C_b \qquad P_a \leq P_b$$

Related Models and Applications

- INDEX (Varaiya et al.)

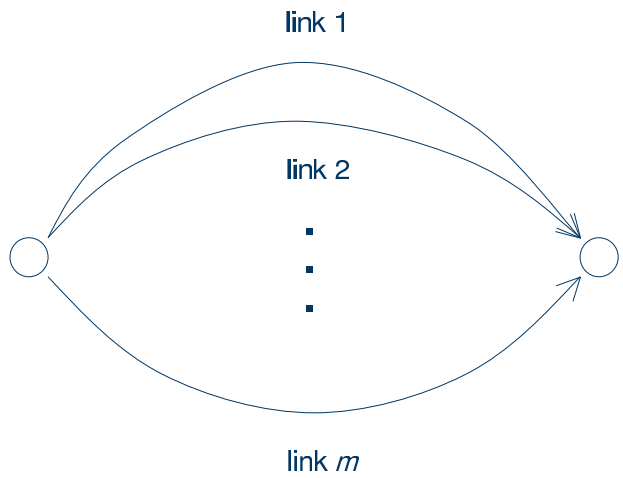
Platinum Service	BW_1	$Price_1$
Gold Service	BW_2	$Price_2$
Silver Service	BW_3	$Price_3$
Bronze Service	BW_4	$Price_4$

→ service class: volume insensitive

→ overprovisioning

Related Models and Applications (cont.)

- Parallel link routing (Orda et al.)



→ Type A, B, C (homogenous users)

→ convex cost functions/utilities

Game-theoretic Structure

- Game with/without pricing
  - influence of pricing
- Splittable/unsplittable
- Stability
  - Conditions for existence of Nash equilibria
  - Characterization of Nash equilibria
  - Properties of Nash equilibria: system and Pareto optimality
- General game structure
- Simple user optimal strategy

Game-theoretic Structure (cont.)

- Stability
  - Nash equilibria, in general, need not exist
  - Quasi-concave (but not concave) utility functions
  - Pricing does not necessarily help
  - Splittable games
  - Nash need not be Pareto nor system optimal
  - Collapse Nash/Pareto/system when “resource plentiful”

  → predictable service if cyclic clique is small

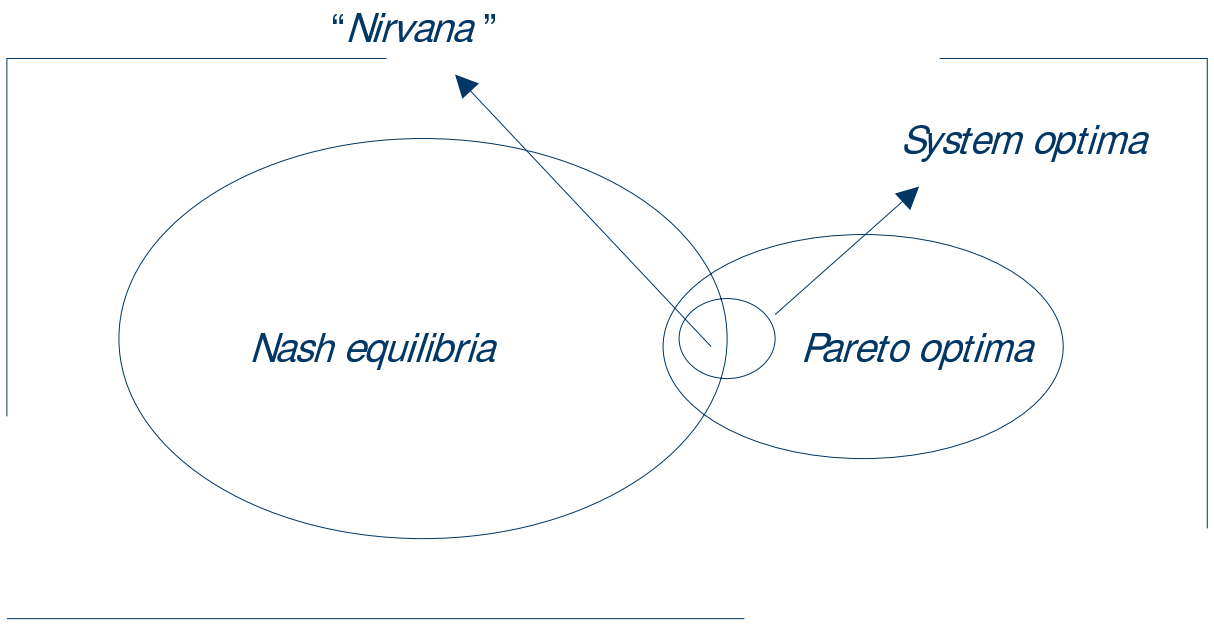
Game-theoretic Structure (cont.)

- Pricing
  - Can cause instability
  - Does help efficiency (Grouping Theorem)
  - Volume control (i.e., variable λ_i); can help stability
  - General pricing

Game-theoretic Structure (cont.)

- General game structure

  nontrivial configuration class separation



Game-theoretic Structure (cont.)

- Self-optimization procedure

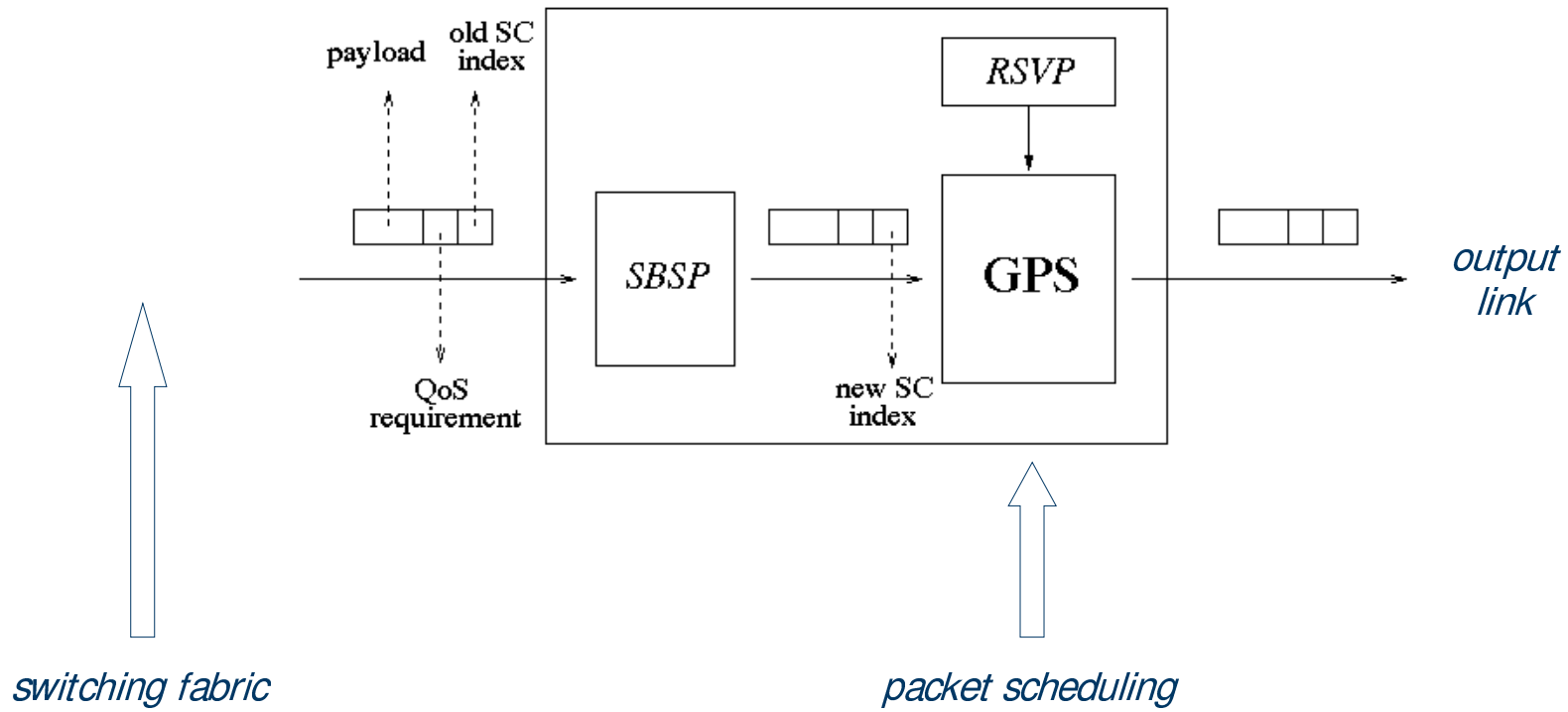
User i : Select service class j such that

1. User i's QoS requirement is satisfied in j
2. j is least cost

→ facilitate selfishness emulation

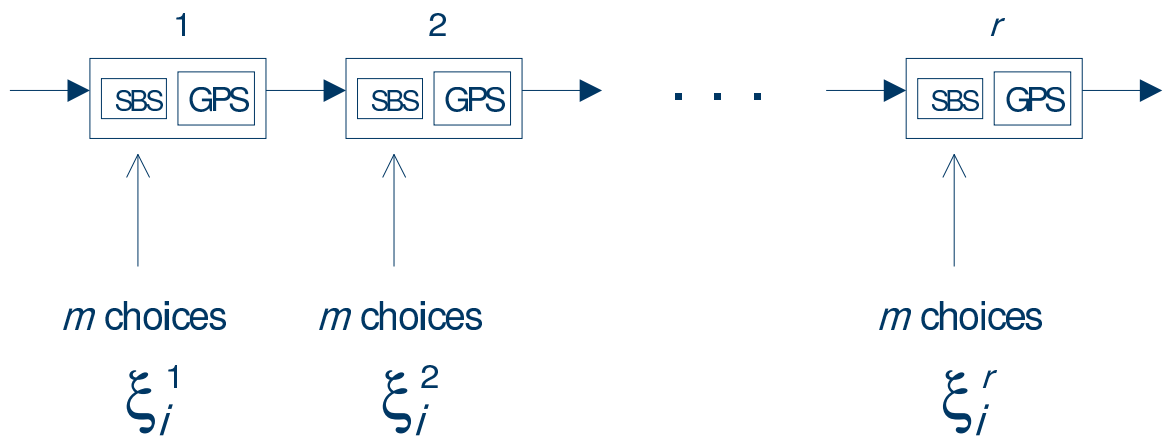
SBS Enabled Router

- Trap routine



Many-switch System

- End-to-end QoS path



$\xi_i^k \in \{0,1,...,m\}$; service class index at switch k

Many-switch System (cont.)

- Optimization problem (user i):

$$\text{minimize} \quad \sum_k \lambda_i^k p_i(\xi_i^k) \longrightarrow \textit{cost}$$

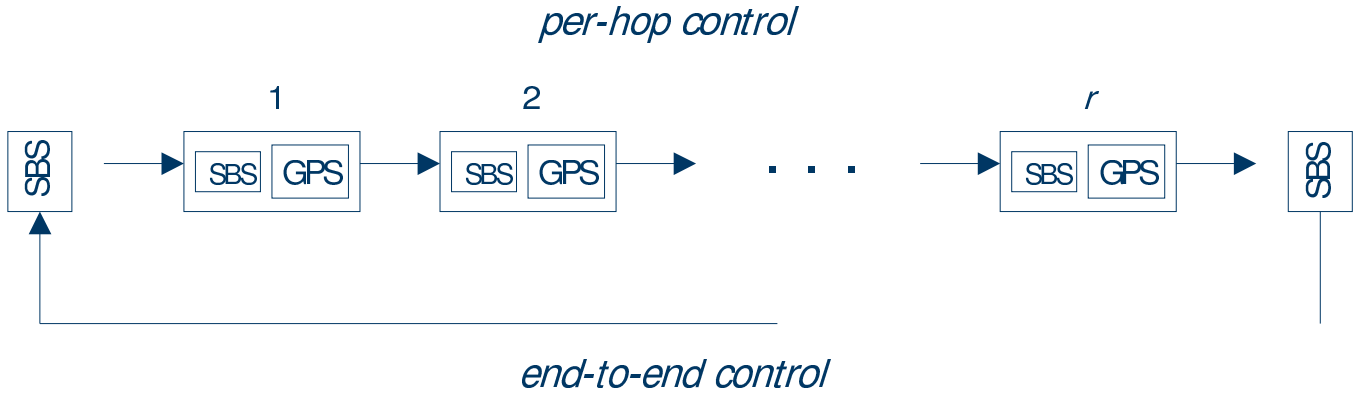
$$\text{subject to} \quad x_i \le \theta_i \longrightarrow \textit{end-to-end QoS}$$

$\rightarrow$ threshold utility functions

$\rightarrow$ single user problem is NP-hard

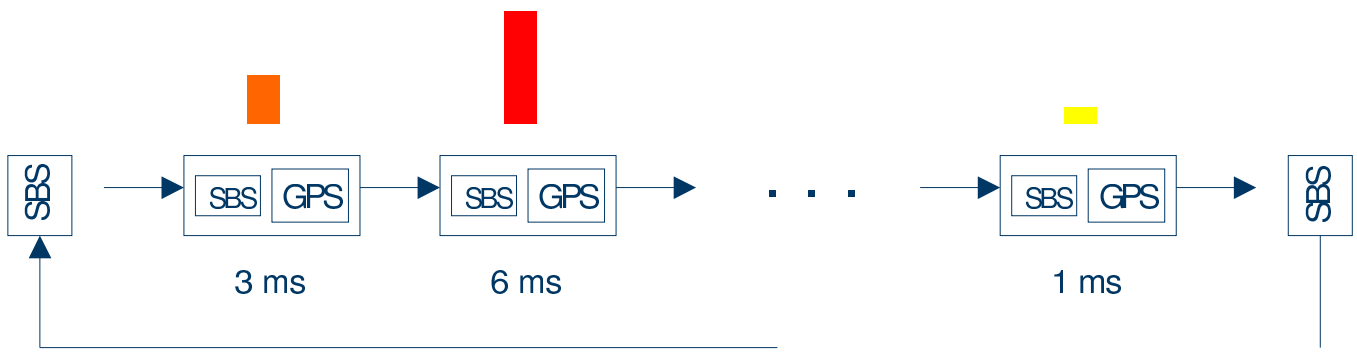
Many-switch System (cont.)

- Solution approaches
  - Single-switch reduction
  - Lagrangian optimization
  - Scalar QoS control



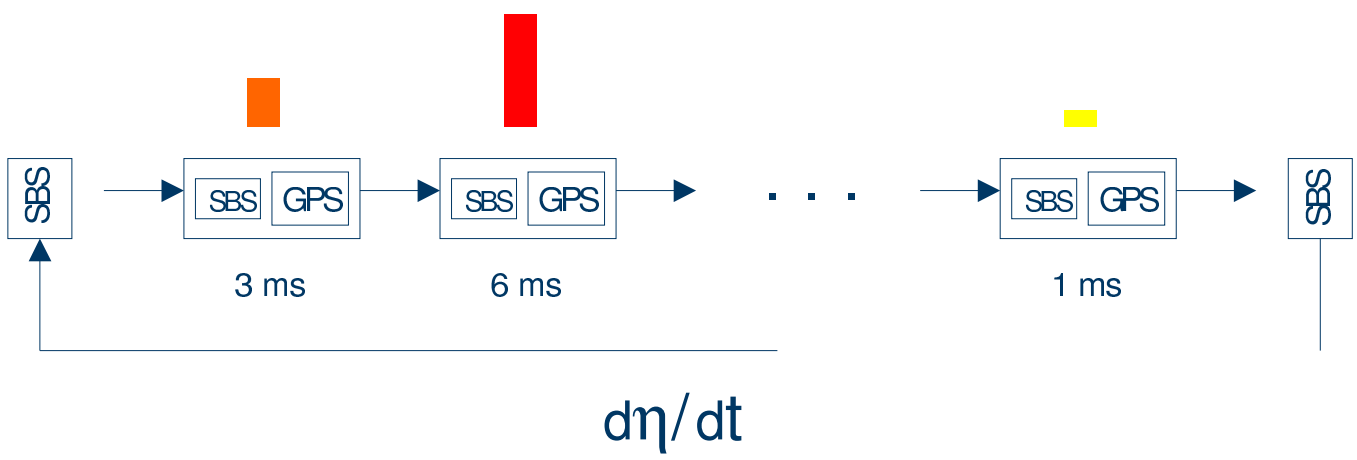
Distributed QoS Control

- Single-switch reduction
- Local QoS responsibility assignment
- Decoupling: single-switch self-optimization
- Load imbalancing
- Constant header: QoS requirement vector + extra bits



Distributed QoS Control (cont.)

- Lagrangian optimization
  - Lagrangian multiplier η
  - Decoupling
  - 2nd order load imbalancing
  - Constant (smaller) header: value of η



Distributed QoS Control (cont.)

Diff-Serv

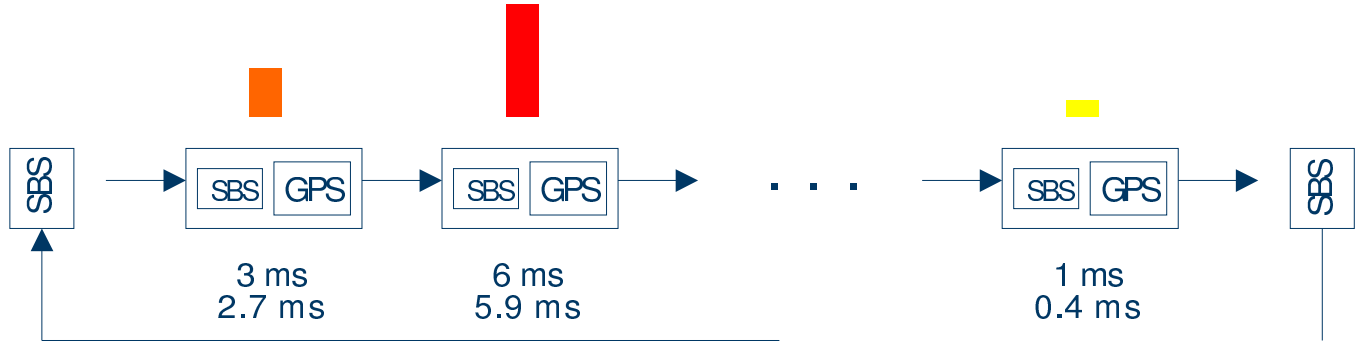
- Scalar QoS control
  - Control parameter η → *DSS*
  - End-to-end control
    - $d\eta_i / dt = f(\theta_i, x_i)$ → *SLA*
    - Per-flow η control at end points (i.e., η_i)
  - Per hop control
    - $\partial c / \partial \eta \leq 0$ → *PHB*
    - Aggregate-flow control: zero per-flow state

Distributed QoS Control (cont.)

- Load imbalancing (1st order)
- Load imbalancing theorem (2nd order)
  - If $\eta\uparrow$ then first switch on underloaded router
  - If $\eta\downarrow$ then first switch on overloaded router

  $\rightarrow$ moves in “proper” direction
  $\rightarrow$ convexity of performance function
  $\rightarrow$ η possesses nonuniform meaning

Distributed QoS Control (cont.)



$d\eta/dt > 0$

Desirable Properties

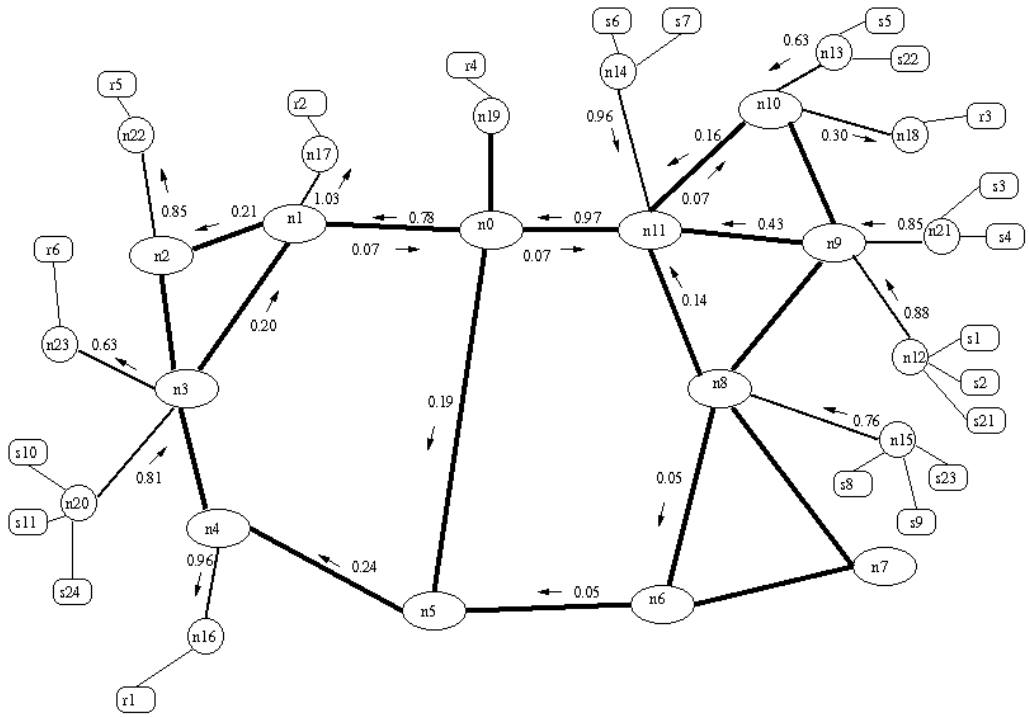
- Well-defined, simple user/network interface
  - User conveys QoS requirement vector
  - η related to price and cost: $S = S(\eta)$; $S' \geq 0$
  - η control: inside or outside the network proper
- QoS-sensitive services (vs. "differentiated" services)
  - Algorithmic or "how" aspect of soft QoS
- Scalability
  - Aggregate flow control
  - Implicit admission control

Desirable Properties (cont.)

- 2-level noncooperativeness w/o direct user control
  - Network behavioral contract: selfishness emulation
  - QoS conveyance, η control, volume control
- Promising results in WAN simulations
  - Predictable, “stable” service
  - Performance comparison

WAN Benchmarking

- vBNS benchmarking



Network Systems Lab

WAN Benchmarking (cont.)

- Traffic configuration

Traffic Flow				
	Flow No.	Src.	Dest.	Rate (Mbps)
Application	1	s1	r2	2.99
	2	s2	r4	8.53
	3	s3	r5	10.24
	4	s4	r5	2.99
	5	s5	r6	3.98
	6	s6	r1	12.05
	7	s7	r4	2.99
	8	s8	r2	2.99
	9	s9	r7	2.99
	10	s10	r2	7.88
	11	s11	r3	2.59
Background	12	s21	r2	2.05
	13	s22	r6	5.85
	14	s23	r4	5.85
	15*	s24	r3	4.81

WAN Benchmarking (cont.)

- SBS Performance

flow	pls. req.	delay req.	SBS		
			pls.	delay	cost
1	.05	.100	.0365	.031	49.64
2	.05	.100	.0341	.019	39.66
3	.05	.100	.0338	.041	59.66
4	.01	.080	.0002	.040	60.00
5	.01	.080	.0003	.045	60.00
6	.05	.100	.0341	.034	49.66
7	.01	.080	.0004	.008	30.00
8	.05	.100	.0363	.032	49.65
9	.01	.080	.0000	.043	50.00
10	.05	.100	.0264	.023	29.74
11	.01	.080	.0000	.045	60.00

WAN Benchmarking (cont.)

- Performance Comparison

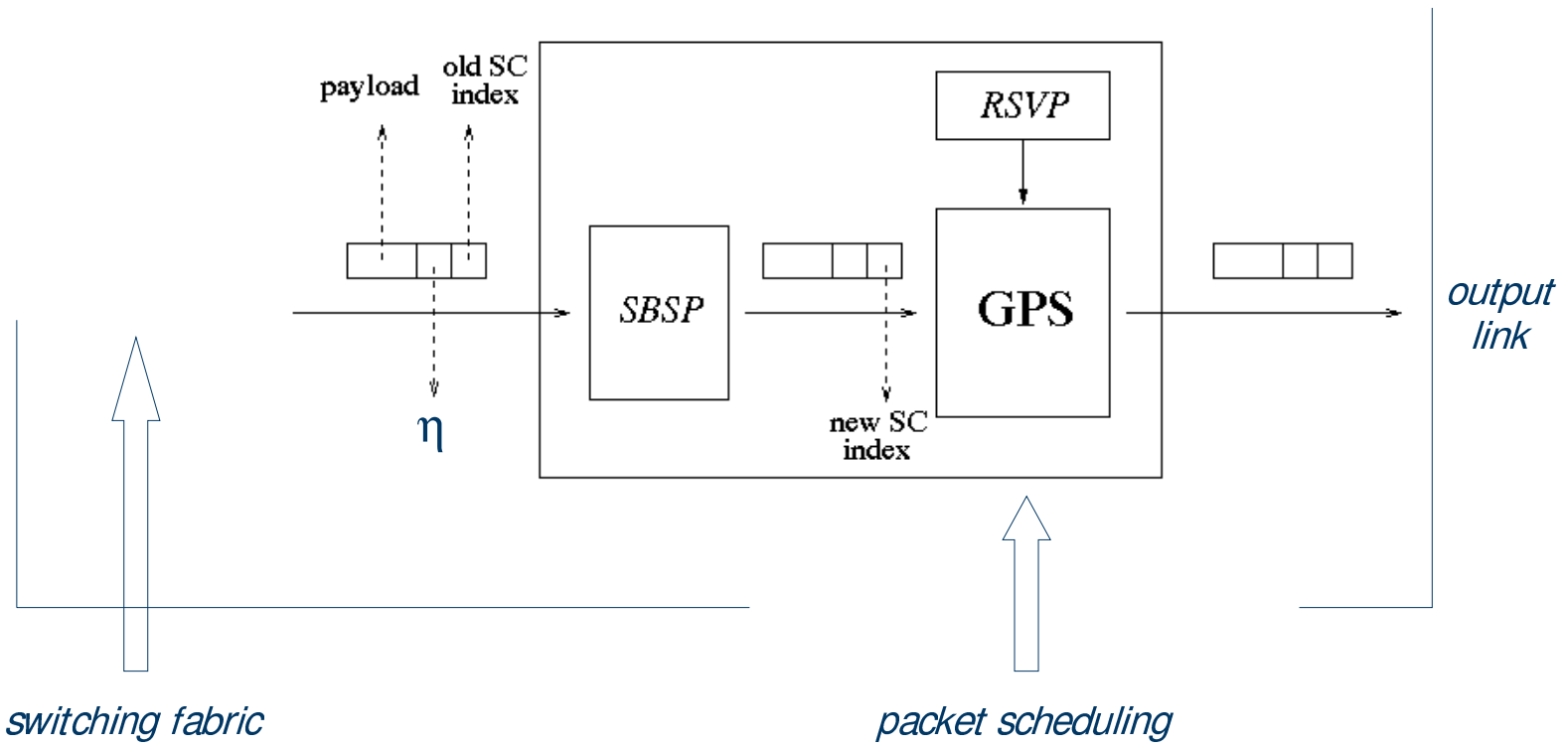
Inst.	Resource Plentiful				In-between				Resource Scarce			
	sbs	fix.	fifo	ran.	sbs	fix.	fifo	ran.	sbs	fix.	fifo	ran.
1	0	1	3	4	1	5	3	6	4	5	3	6
2	0	1	2	1	0	6	2	4	1	6	2	4
3	0	1	4	1	2	5	4	2	2	5	4	4
4	0	0	2	3	3	0	2	3	3	6	3	5
5	0	0	2	4	3	1	3	4	4	2	4	4
6	0	1	2	4	2	2	2	5	3	2	4	5

Desirable Properties (cont.)

- Implementability
  - Can be expressed within IETF Diff-Serv language
  - IPv4 based implementation
  - Cisco 72xx series router customization of SBS
  - “Purdue Infobahn” testbed

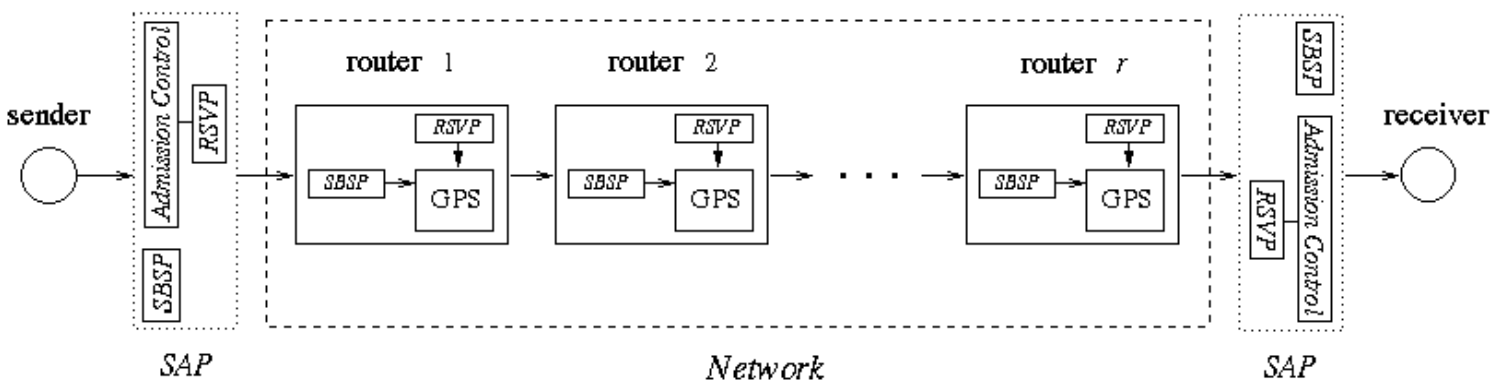
SBS Enabled Router

- Trap routine



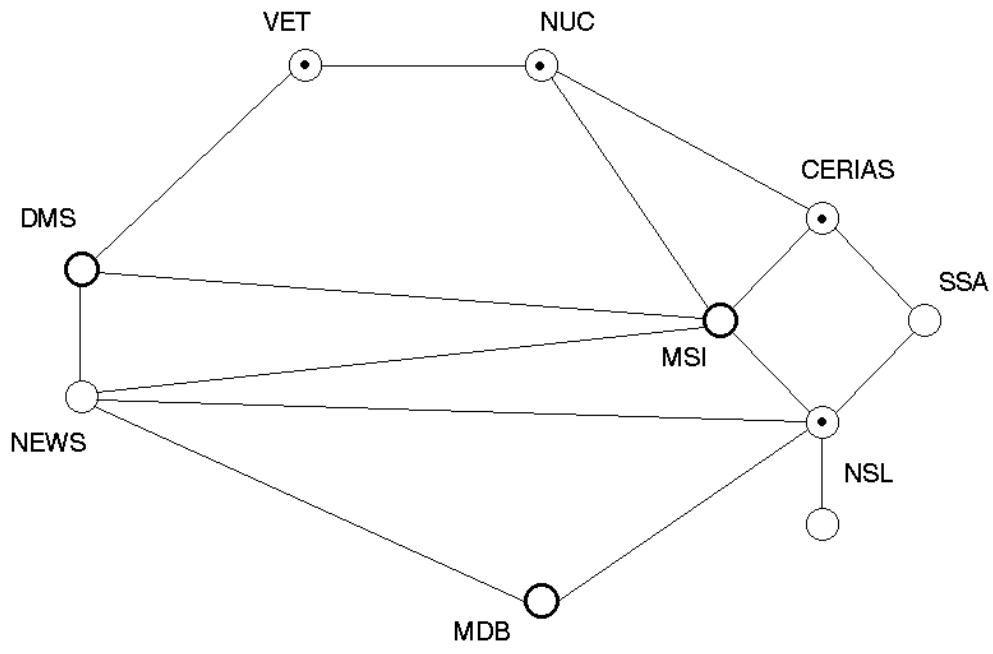
GS + SBS Architecture

- GS + SBS control path



Purdue Infobahn Testbed

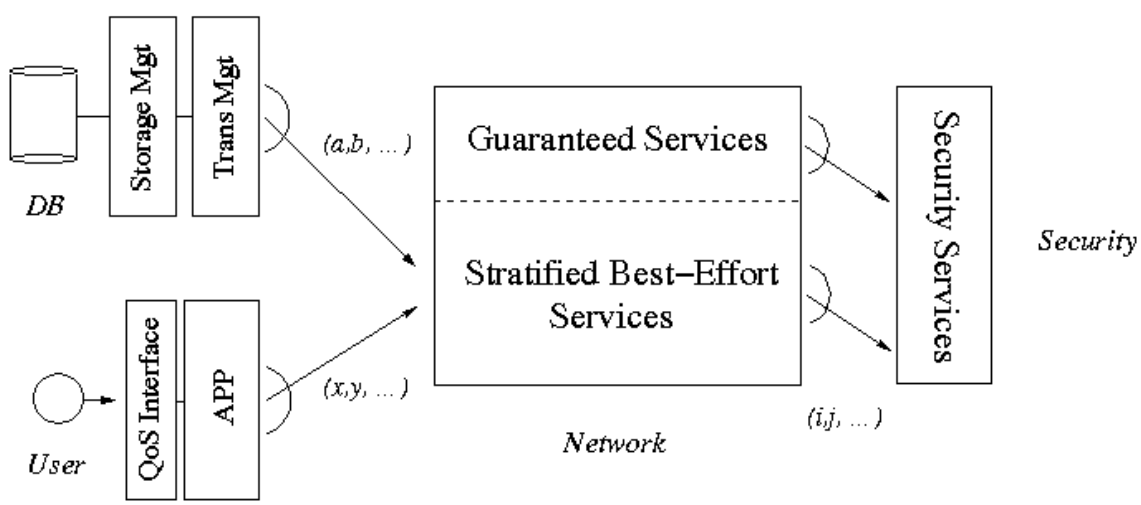
- Internetwork topology



Applications

- Multimedia DB, distance learning, e-commerce, ...

QoS network substrate & interface:



On-going and Planned Activities

- SBS implementation & benchmarking 1~2 years
- Benchmarking with multimedia apps
- Performance comparisons with other Diff-Serv arch.
- IETF Diff-Serv draft
- Game-theoretic work non-concave utilities
- Burstiness and short connection management
- LRD and long connection management
- End-to-end QoS amplification
- Distributed QoS control