Workshop on Reducing Internet Latency
goals for taxonomy session

- survey sources of latency
- categorise solutions
  - quantify benefits
  - consider deployment aspects
  - short-term & long-term applicability

- common reference framework for discussions

- schedule
  - [10-15] Joe Touch, ISI Factors underlying the problem space
  - [10-15] **Bob Briscoe**, BT Solution space – systems focus
  - [10-15] Lucien Avramov, Cisco Solution space – intra-box focus
  - [10-15] open to contributions from the floor
  - [50-30] discussion

REDUCING INTERNET TRANSPORT LATENCY

survey of latency reducing techniques and their merits

a work in progress

Bob Briscoe, Anna Brunstrom, Gorry Fairhurst, Stein Gjessing, David Hayes, Andreas Petlund, David Ros, Ing-Jyh Tsang

goal for this talk

- industry roadmap of techniques
- gain vs pain
  - latency reduction against deployability

- "A Survey of Latency Reducing Techniques and their Merits"
  - ~190 references
  - a work in progress
  - available soon via http://riteproject.eu/publications/

- evolved from BT roadmap work, but repurposed
  - a company tries to prioritise the quick wins
  - an industry also needs to identify hard problems being avoided

latency-reducing techniques

organised by sources of delay

3.1 Structural delays

3.1.1 Server placement
3.1.2 Sub-optimal route latency
3.1.3 Name resolution
3.1.4 Content placement

3.2 Interaction between endpoints

3.2.1 Protocol Initialisation
3.2.2 Secure session initialisation
3.2.3 Packet loss recovery delays:

3.3 Reducing delays along transmission paths

3.3.1 Propagation delay
3.3.2 Switching/routing delay
3.3.3 Queueing delay
3.3.4 Error correction delays

3.4 Reducing delays related to link capacities

3.4.1 Insufficient capacity
3.4.2 Redundant information
3.4.3 Under-utilised capacity
3.4.4 Collateral damage
3.4.4 Medium acquisition delays

3.5 Intra-end-host delays

3.5.1 Transport Protocol Stack buffering
3.5.2 Operating system delay

latency-reducing techniques

organised by sources of delay

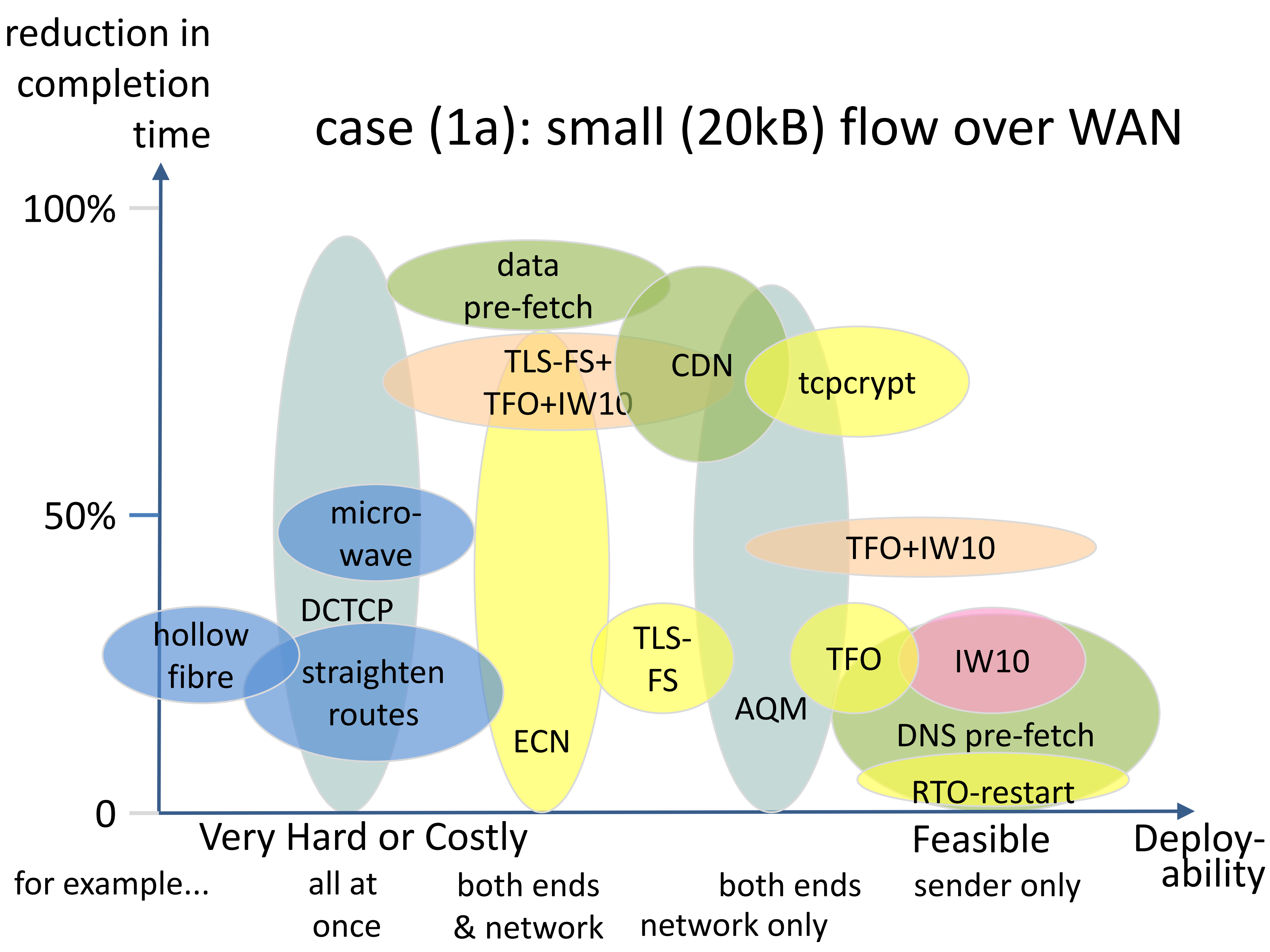
3.1 Structural delays
- 3.1.1 Server placement
- 3.1.2 Sub-optimal route latency
- 3.1.3 Name resolution
  - 3.1.3.1 DNS cache placement
  - 3.1.3.2 DNS cache pre-fetching
- 3.1.4 Content placement
  - 3.1.4.1 Proxies and caches
  - 3.1.4.2 Prediction and latency hiding

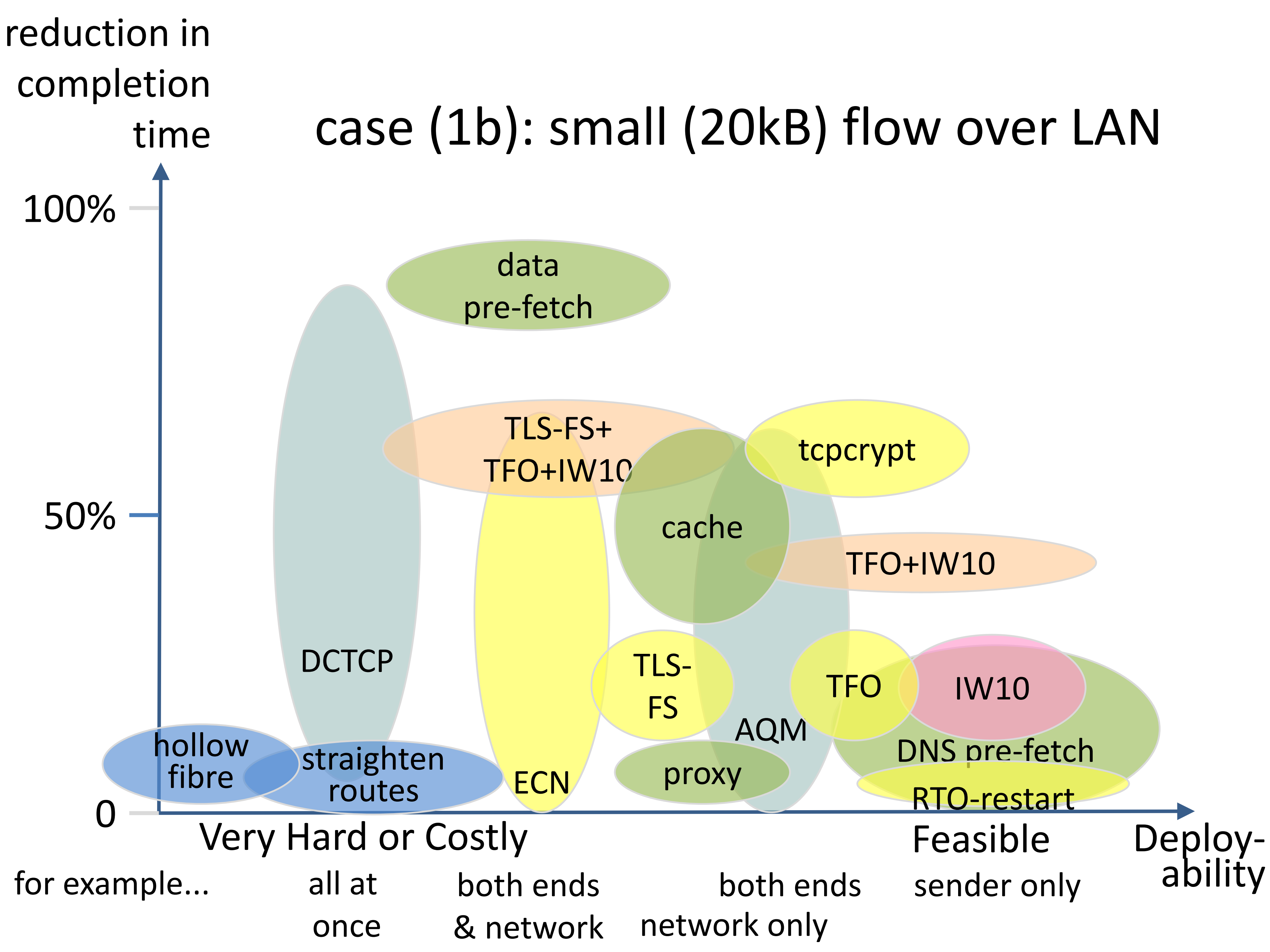
3.2 Interaction between endpoints
- 3.2.1 Protocol Initialisation
  - 3.2.1.1 TCP fast open
  - 3.2.1.2 Pipelining
- 3.2.2 Secure session initialisation
  - 3.2.2.1 Transport layer security negotiation
  - 3.2.2.2 Building encryption into TCP
  - 3.2.2.3 Bootstrapping security from the DNS
- 3.2.3 Packet loss recovery delays:
  - 3.2.3.1 Application tolerance to errors and order of delivery
  - 3.2.3.2 Reduce error detection time
  - 3.2.3.3 Add redundancy

3.3 Reducing delays along transmission paths
- 3.3.1 Propagation delay
  - 3.3.1.1 Straighter cable paths
  - 3.3.1.2 Higher signal velocity
  - 3.3.1.3 Combining higher signal velocity and straighter routes
- 3.3.2 Switching/routing delay
- 3.3.3 Queueing delay
  - 3.3.3.1 Flow and circuit provisioning
  - 3.3.3.2 Packet scheduling
  - 3.3.3.3 Traffic shaping and policing
  - 3.3.3.4 Small buffers
  - 3.3.3.5 Queue management
  - 3.3.3.6 Transport-based queue control
- 3.3.4 Error correction delays
  - 3.3.4.1 Improve channel quality
  - 3.3.4.2 Hop based error correction and packet ordering

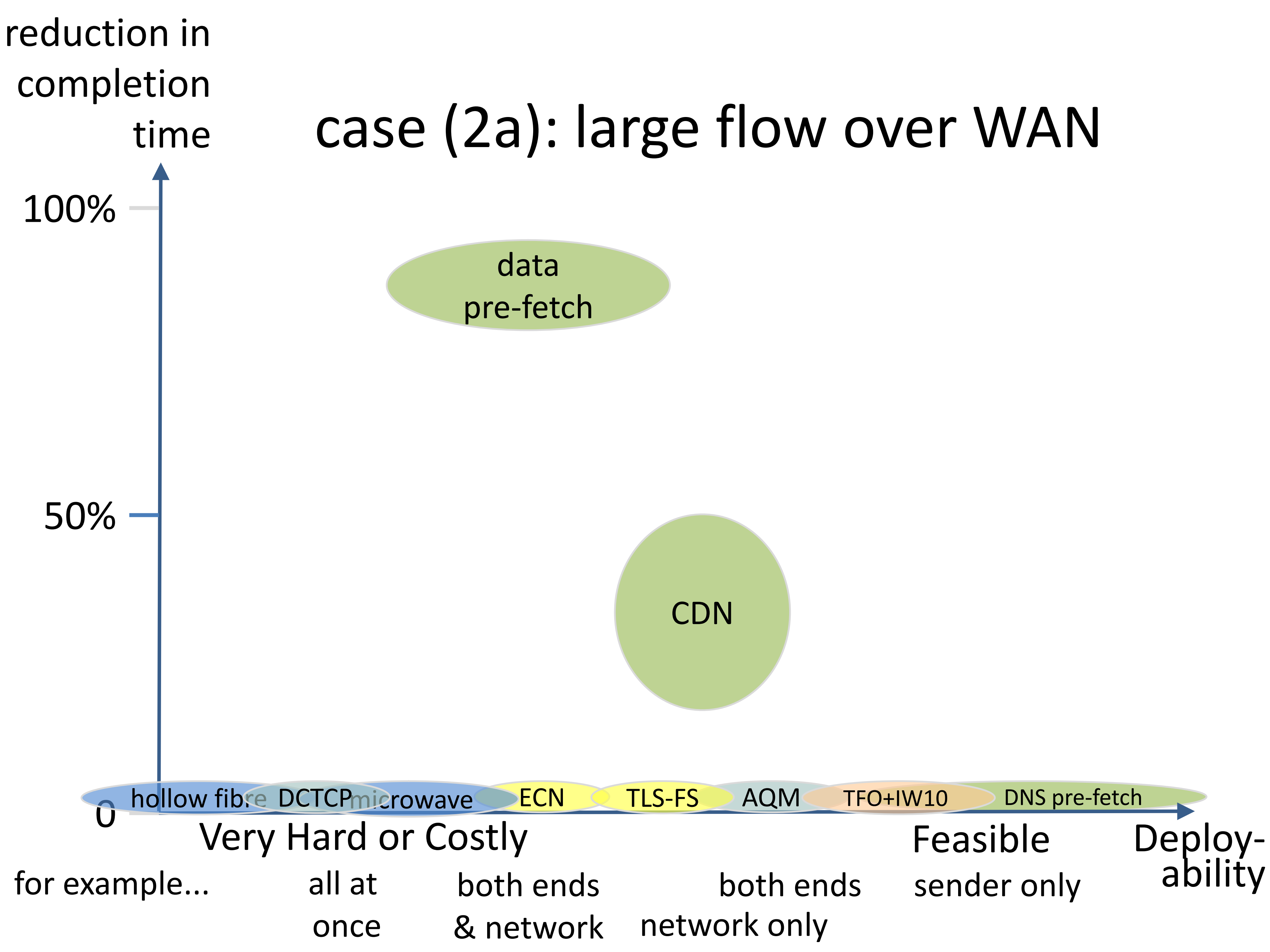
3.4 Reducing delays related to link capacities
- 3.4.1 Insufficient capacity
- 3.4.2 Redundant information
- 3.4.3 Under-utilised capacity
  - 3.4.3.1 More aggressive congestion control
  - 3.4.3.3 Rapidly sensing available capacity
- 3.4.4 Collateral damage
- 3.4.4 Medium acquisition delays

3.5 Intra-end-host delays
- 3.5.1 Transport Protocol Stack buffering
- 3.5.2 Operating system delay


reduction in completion time
case (1a): small (20kB) flow over WAN
100%
50%
0
data pre-fetch
TLS-FS+ TFO+IW10
CDN
tcpcrypt
micro-wave
DCTCP
TFO+IW10
hollow fibre
straighten routes
TLS-FS
TFO
IW10
AQM
ECN
DNS pre-fetch
RTO-restart
Very Hard or Costly
Feasible
Deploy-ability
for example...
all at once
both ends & network
both ends
network only
sender only

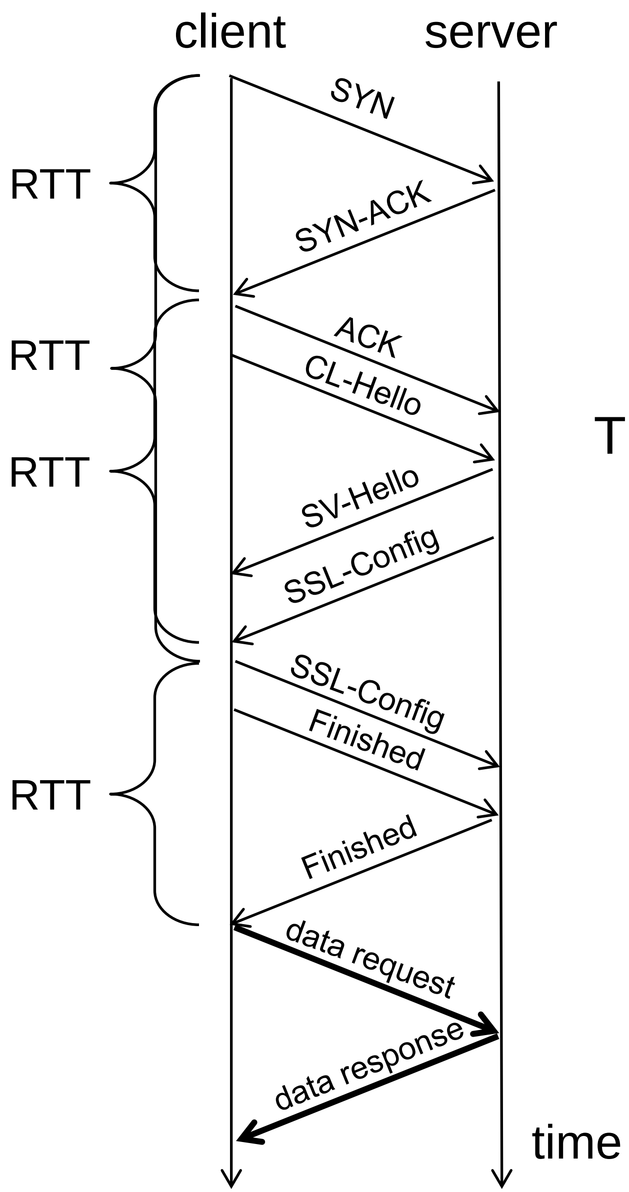

case (1b): small (20kB) flow over LAN
reduction in completion time
100%
50%
0
data pre-fetch
DCTCP
TLS-FS+ TFO+IW10
tcpcrypt
cache
TFO+IW10
TLS-FS
TFO
IW10
AQM
DNS pre-fetch
RTO-restart
hollow fibre
straighten routes
ECN
proxy
Very Hard or Costly
Feasible
Deploy-ability
for example...
all at once
both ends & network
both ends
network only
sender only

case (2a): large flow over WAN



Transaction Layer Security (TLS)

aka secure sockets layer (SSL) or https



TLS adds: 2 RTTs

False Start cuts this to: 1 RTT

TLS with TCP handshake: 3 RTTs

with TCP Fast Open: 2 RTTs

REDUCING INTERNET TRANSPORT LATENCY

a figure of merit: average rate

Bob Briscoe, BT

Anna Brunstrom, Mohammad Rajiullah, Karlstad University

Olga Bondarenko, Simula Research Labs

inaccessible capacity in a dedicated access link

capacity inaccessible to a lone flow in a:

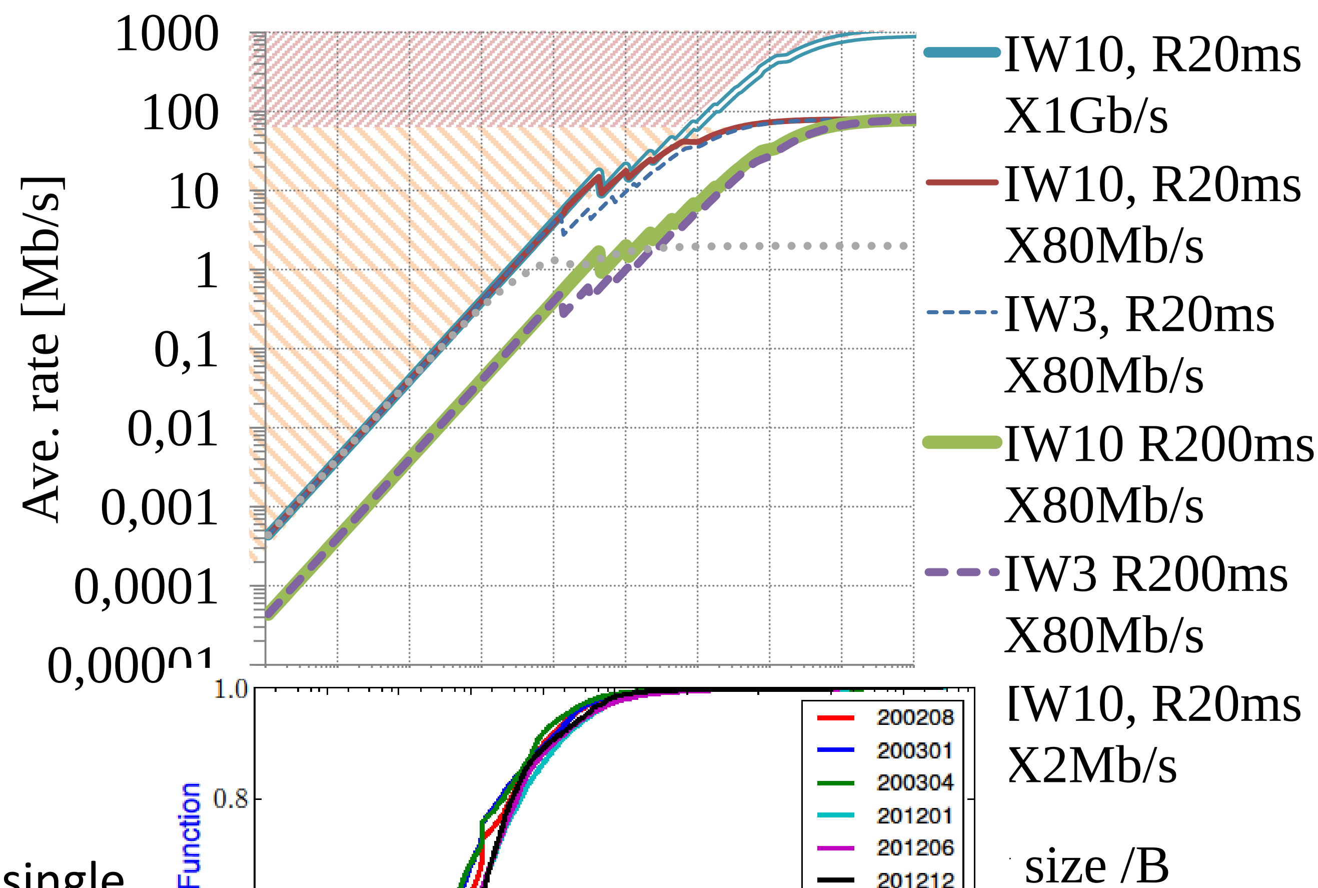
1Gb/s link

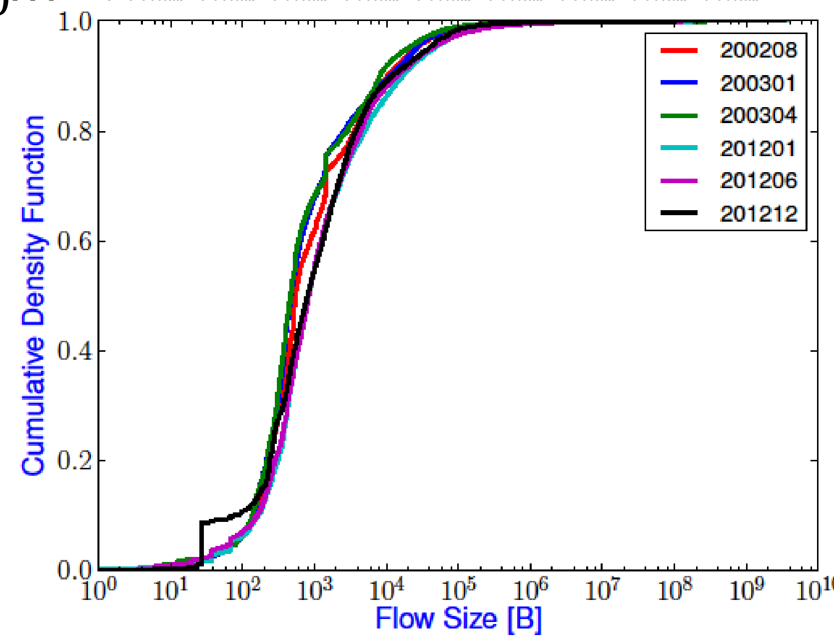
80Mb/s link

IW: initial window (Google has just increased from 3 to 10)

R: round trip time (~20ms: intra-UK ~200ms inter-continent)

X: bottleneck capacity





- below a certain size, a single transfer (of any number of flows) is limited by the slow-start rule, not capacity
- as capacity growth continues, more transfers are limited by the rule than by capacity

CDF w.r.t # of Flows

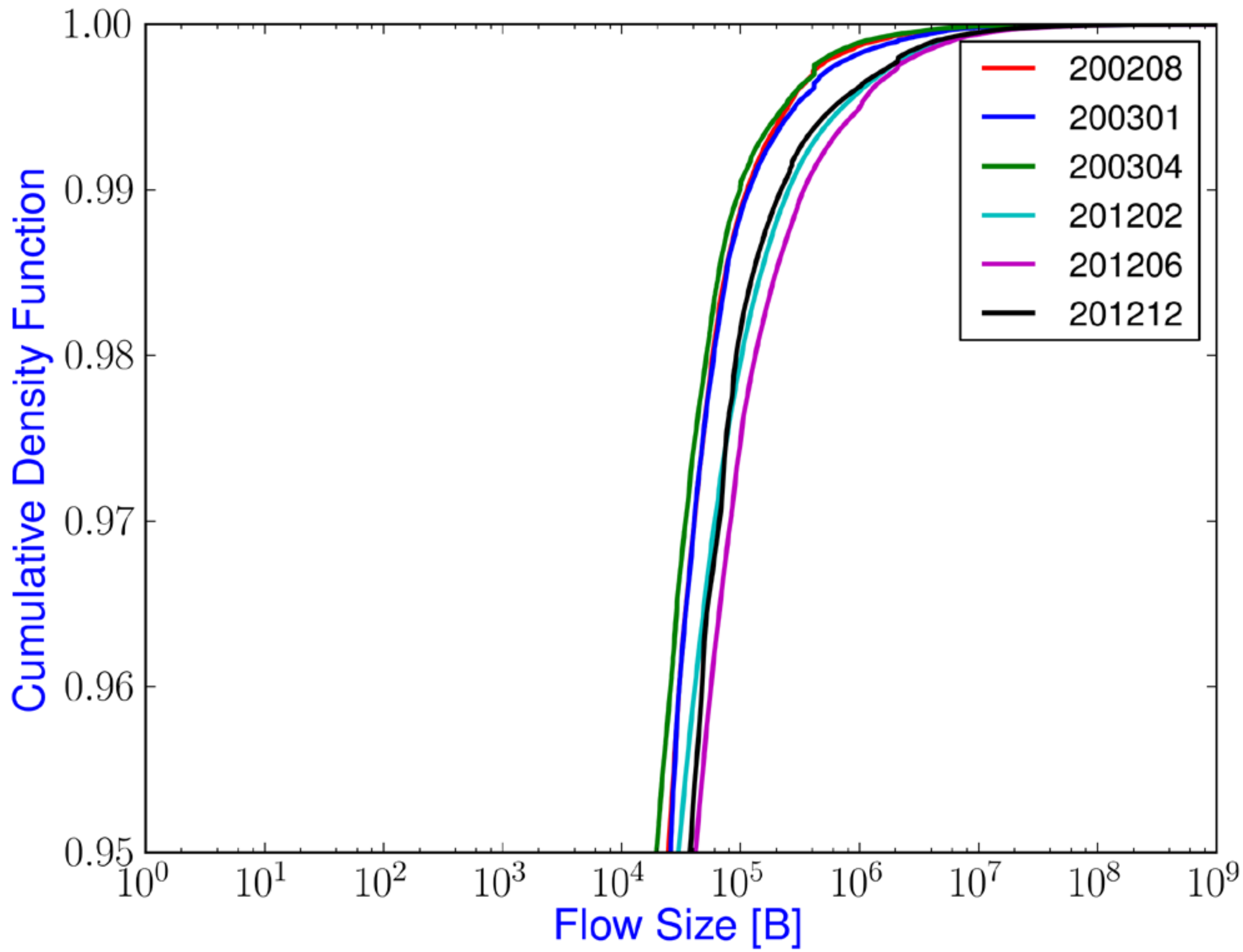


Fig. 1: Prob. of number of flows seen for a given flow size

CDF w.r.t Fraction of Bytes

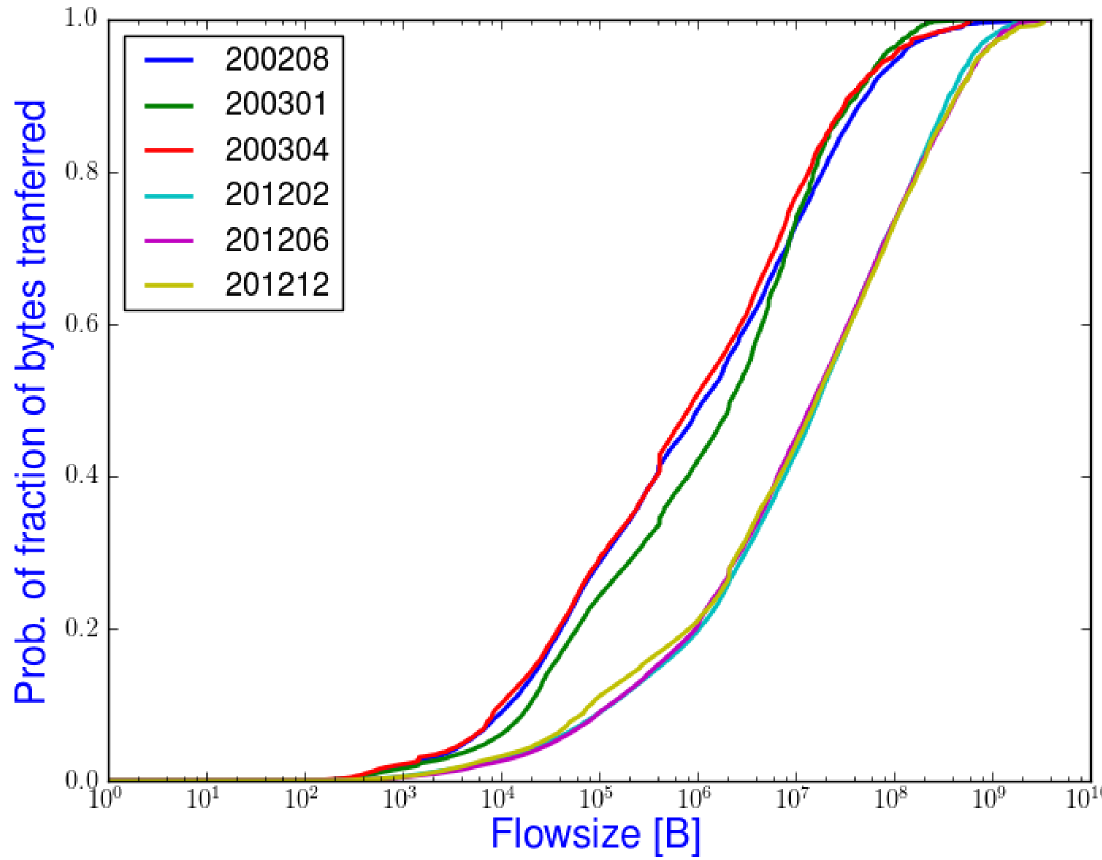


Fig. 2: Prob. of fraction of total bytes transferred for a given flow size