On improving data transmission in networks

Eugen Dedu

Maître de conférences
Research: Institut FEMTO-ST, DISC department
Teaching: Univ. de Franche-Comté, IUT de Belfort-Montbéliard

Habilitation defense
Montbéliard, France
3 dec. 2014

http://eugen.dedu.free.fr
eugen.dedu@univ-fcomte.fr

News since 7/10/2014 manuscript

- Paper to IEEE UIC conference accepted
- Paper submitted and accepted to IEEE Aerospace Conference
- 1 week of staying in USA in communication in nanonetworks, article being written
- RGE research regional meeting organisation in Montbéliard (gathering all researchers in computer networks in East of France)

Plan

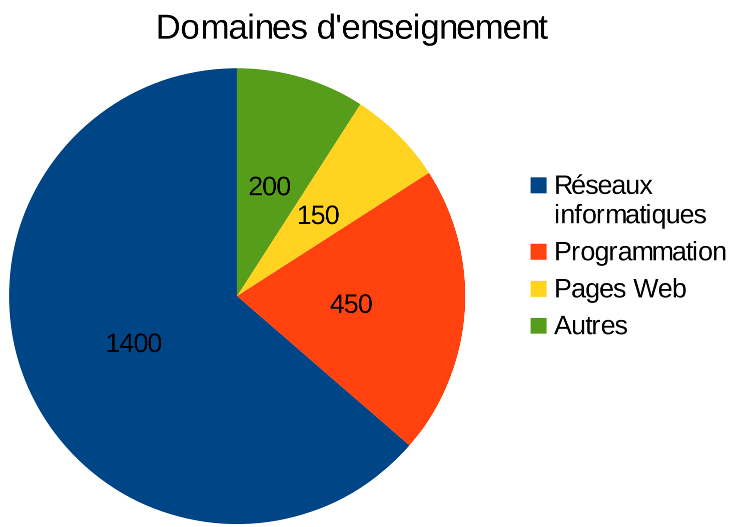
- Short CV (in French)
- 1. Congestion control in networks
- 2. Adaptive video streaming with congestion control
- 3. Communication in distributed intelligent MEMS
- 4. Communication in wireless nanonetworks
- Conclusions and perspectives

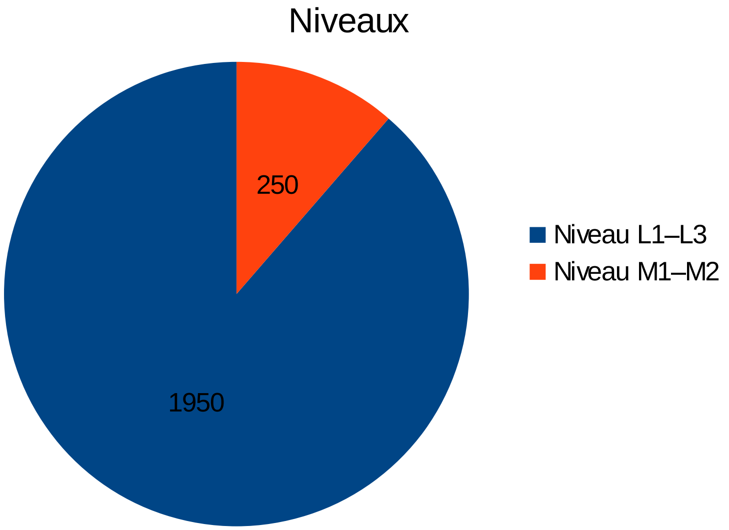
Expériences professionnelles

- 1993–1998 Diplôme d'ingénieur, informatique, Bucarest, Roumanie
- 1997–1998 M2 recherche (DEA), systèmes distribués, Toulouse
- 1998–2002 Thèse de doctorat, parallélisation de systèmes multi-agent, Versailles/Metz
- 2002–2003 ATER, parallélisation de systèmes multi-agent, Versailles
- 2003–présent, Maître de conférences, réseaux informatiques, Montbéliard <= **je détaille que cette partie**

Activités pédagogiques

- IUT de Belfort-Montbéliard, département Réseaux et Télécommunications
- Porteur du dossier et ex-responsable de la licence professionnelle « Chargé d'affaires en R&T » (2006–2011)
- Participation à des activités variées du département : site Web, organisation WAN, présentation aux lycées, entretiens avec les candidats, forums, portes ouvertes et beaucoup d'autres
- Élu dans le conseil de l'IUT et conseil restreint (2010–2014)





Activités de recherche

Public. intern. (21 réf, 4 non réf)

	CC	Vidéo	diMEMS	Nano	Total
J	1	4	2	0	7
C	5	3	8	2	18

	...'09	'10	'11	'12	'13	'14	'15	Total	
J		1	1	1	3	1		7	1.5/an
C	7	3	1	3	1	2	1	18	2/an

Projets

Rôle	Type	Financement
PI	Région	160 k€
Task leader	ANR intern.	500 k€
Membre	ANR intern.	440 k€

Co-porteur 1 dossier BQR,
1 dossier de bourse de thèse Région

Encadrement doctoral

Doctorant	Co-encadrement	Domaine	Soutenance	Poste actuel
M. A. Zainuddin	50 %	Nano	2ème année	
H. Skima	30 %	diMEMS	2ème année	
A. Habibi	20 %	diMEMS	—	
W. Ramadan	70 %	CC + Vidéo	2011	MdC Syrie
K. Boutoustous	70 %	diMEMS	2009	R&D entreprise
S. Linck	60 %	CC	2008	Ch. contr. Reims

3 M2 recherche et 3 M2 pro/stage ingénieur, encadrement à 100%

Rayonnement scientifique

- 9 fois program vice-chair de conf. int.
- 28 fois membre du comité technique conf. int.
- 11 reviews pour journaux int.
- 3 fois membre du comité d'organisation de conf. int.
- Organisateur de la réunion RGE oct. 2014
- Dans mon laboratoire :
  - 2012–présent : Membre du Conseil d'Orientation Scientifique
  - 2008, 2010 : Membre du comité de sélection des MdC
  - 2006–2007 : Membre du conseil du laboratoire

Rayonnement grand public / Développement

- 2009–présent : Développeur du logiciel ekiga (vidéoconférence) :
  - 500 commits, 400 bugs fermés, release manager (10 dernières releases), documentation
  - j'interviens aussi dans les deux bibliothèques afférentes, ptlib (devices, multi-plate-forme) et opal (SIP, H323, codecs) : 100 commits
- 2010–présent : Debian Maintainer
  - en charge des paquets ekiga, ptlib et opal
- SLOC : ekiga 100k, ptlib 250k, opal 650k

Research plan

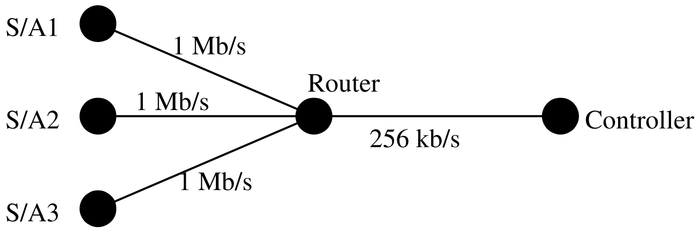
- 2003: fields of research of the lab were: network protocols, especially wi-fi, and video transmission
  - 1. congestion control
  - 2. video transmission, adaptation
- 2006: ANR-funded project Smart surface
  - 3. communication in distributed intelligent MEMS
- 2013: collaboration with USA, Tb/s communication
  - 4. communication in nanonetworks
- In the remaining of the talk I will present my work on these 4 fields through some of the ideas/papers I was co-author of

1.1 Congestion control in networks Sensor networks

G. Bise, M2 student

Problem: we read everywhere that CC is better than no CC
Goal: study CC in centralised control systems / sensor networks
Methodology: Compare UDP and various CC. Does CC bring any benefit?

Simulation topology:
Each sensor sends 1 kB each 50 ms
=> small congestion on right link
All sensors use (1) UDP, (2) TCP, (3) TFRC



Protocol	Sensor	Packets				Delay	
		generated	lost on sensor	lost on network	received	highest	average
UDP	1	7199	0	7163	36	1.61	1.35
	2	7197	0	3432	3765		
	3	7195	0	0	7195		
TCP	1	7199	4425	0 (26 retr)	2774	1.68	1.38
	2	7197	4380	0 (26 retr)	2817		
	3	7195	1577	0 (11 retr)	5618		
DCCP/TFRC	1	7199	3496	60	3643	1.63	1.41
	2	7197	3184	54	3959		
	3	7195	3193	60	3942		

Conclusions:
- In UDP, some sensors can be muted (synchronisation issues caused by DropTail use)
- **Surprisingly**, same amount of packets received, and similar delay
- If congestion (throughput > bandwidth), UDP loses pkts on network, CC protocols on sender => CC does NOT increases throughput, it just smooths it
- In Internet, flows (dis)appear randomly; in sensor networks, data is generated regularly
- If no congestion, CC == no CC

1.2 Congestion control in networks Loss differentiation 1/3

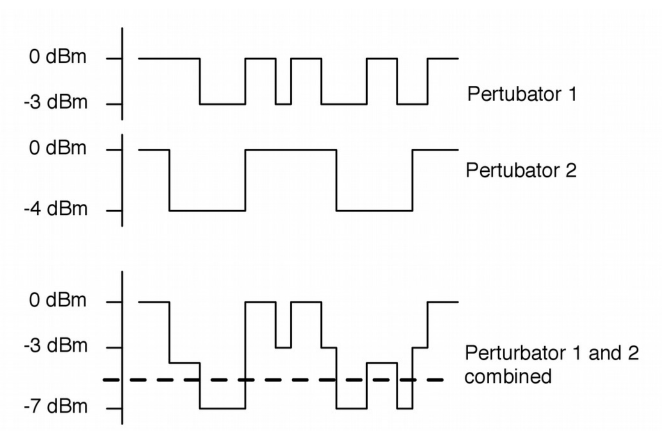
W. Ramadan, PhD student

Problem: transport protocols reduce throughput upon a wireless loss, which is wrong because such loss is not due to congestion

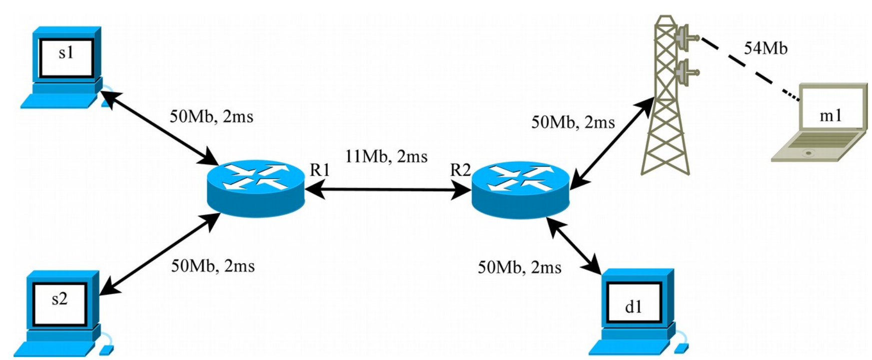
Goal: allow senders to differentiate between congestion (wired) and wireless losses, so that they reduce throughput only for congestion losses

Shadowing-pattern **propagation and loss model:**

- various perturbators can be defined
- perturbators have cumulative effects
- we used 7 perturbators



Network topology in NS2:
1 DCCP/TFRC-like flow from s1 to m1



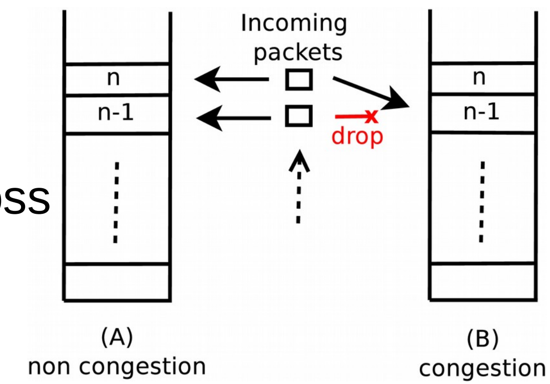
1.2 Congestion control in networks Loss differentiation 2/3

W. Ramadan, PhD student

Influence of losses on RTT

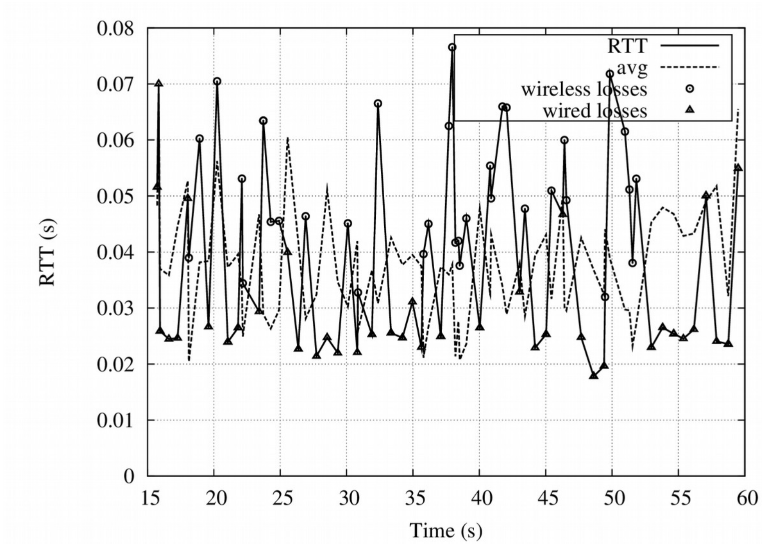
In theory

Congestion loss:
The RTT of the pkt following a congestion loss is smaller than normally

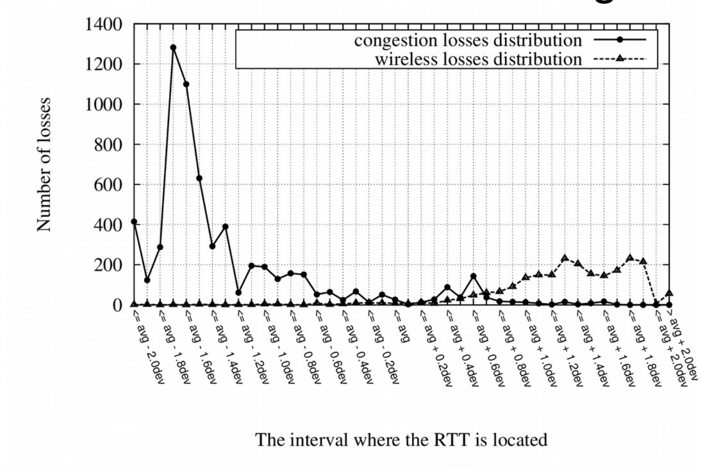


Wireless loss:
The RTT is greater than normally, because a wireless loss appears after 7 retransmissions (losing a packet takes time)

In simulation, same trend as in theory



Choice of threshold, avg+0.6dev

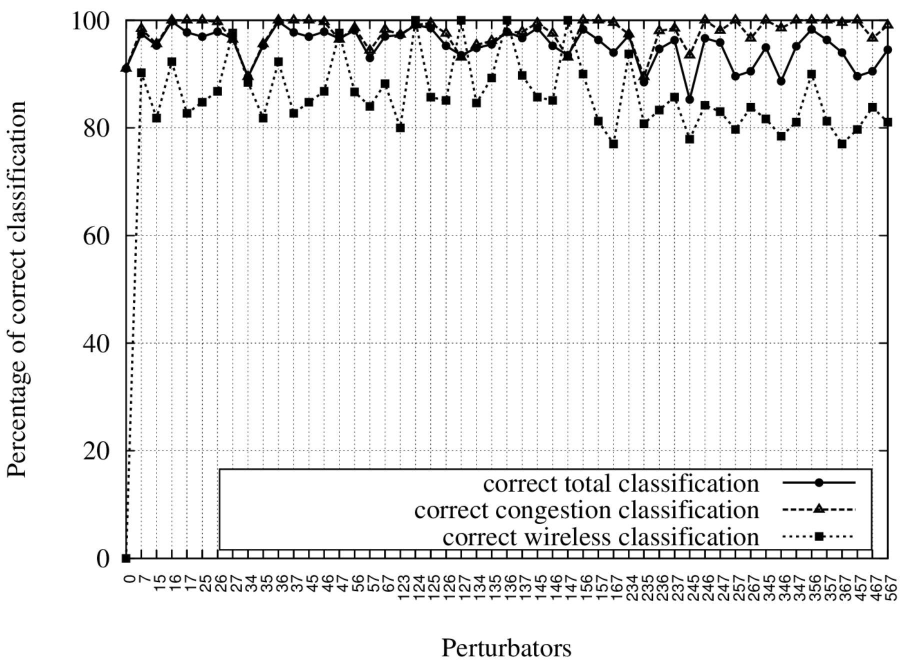


1.2 Congestion control in networks Loss differentiation 3/3

W. Ramadan, PhD student

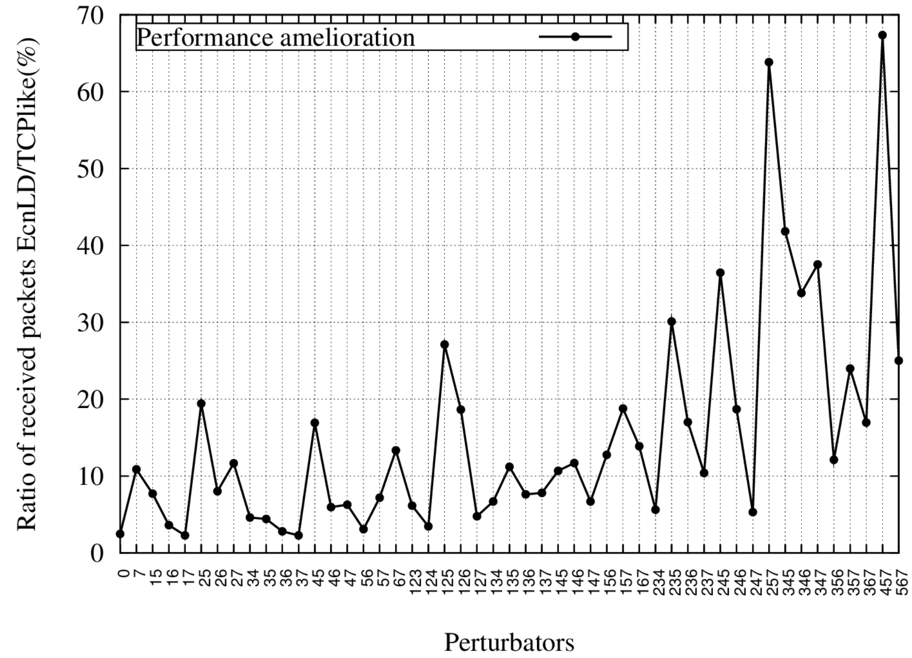
RELD formula:
A loss is due to congestion iff for the following pkt:
ecn > 0 or (n > 0 and RTT < avg + 0.6*dev)

RELD classification accuracy:



Classification accuracy of 92% in average
Congestion losses are better classified
than wireless losses

Comparison with DCCP/TCP-like:



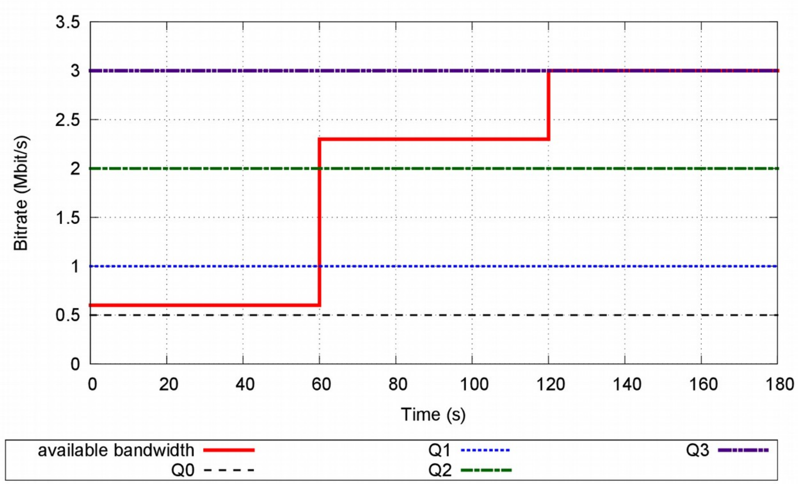
General conclusion:
RELD loss differentiation leads
to more received pkts

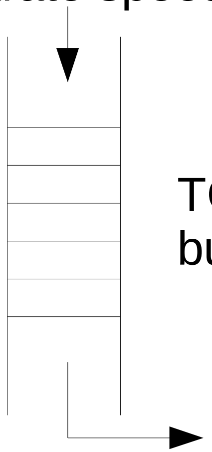
2.1 Adaptive video streaming with CC A video adaptation algorithm 1/2

W. Ramadan, PhD student

Use case: A same video is encoded in several bitrates (0.5, 1, 2, and 3 Mb/s)
Adaptation means switching video bitrate on-the-fly depending on network available bandwidth

Advantage of video adaptation over static encoding





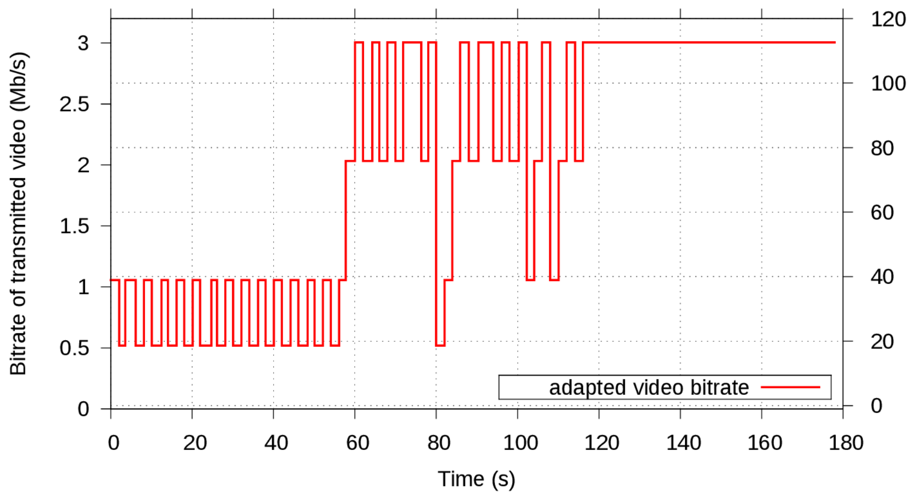
Idea: switch video bitrate according to buffer size
Algorithm:
Each period of 2 sec.:
if write_failure == 0, choose next higher quality
if write_failure < 5%, maintain quality
elsewhere, choose lower quality q' < q(1-write_failure)

2.2 Adaptive video streaming with CC
Quality oscillation avoidance

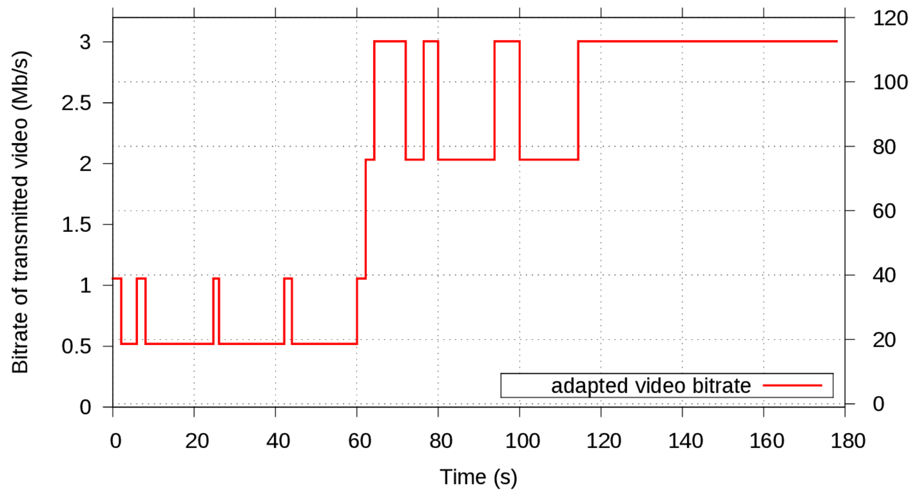
W. Ramadan, PhD student

Problem: continuous quality oscillation, see graph below
Solution: attach to each bitrate a successfulness value,
this value is updated each period of 2 sec. using an EWMA algorithm:
  Si = (1-a)Si + sa
Si, successfulness of bitrate i, between 0 and 1
s, current successfulness
a, weight given to history
Summary: a bitrate which has lead to losses has a small successfulness value
If the adaptation algorithm considers to increase bitrate, it is NOT increased if Si > 0.7

Original: many oscillations



With quality oscillation avoidance



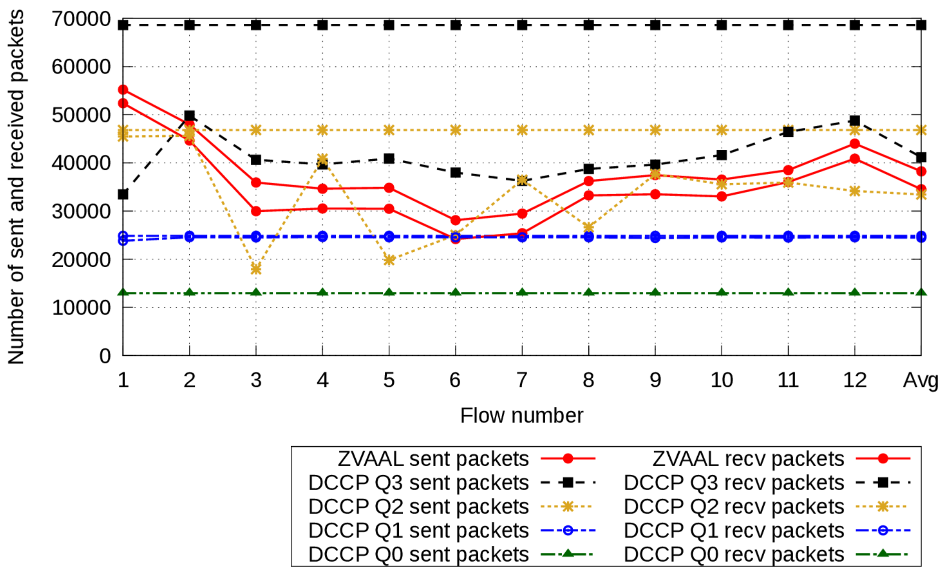
2.1 Adaptive video streaming with CC
A video adaptation algorithm 2/2

W. Ramadan, PhD student

We implemented adaptation with oscillation avoidance on GNU/Linux using DCCP

Comparison of our method to static encoding (without adaptation)

- 12 concurrent flows
- available bandwidth decreases from 1 to 7 and increases from 7 to 12



Out method adapts to the bandwidth
Other methods either lose many packets,
or underuse the network capacity

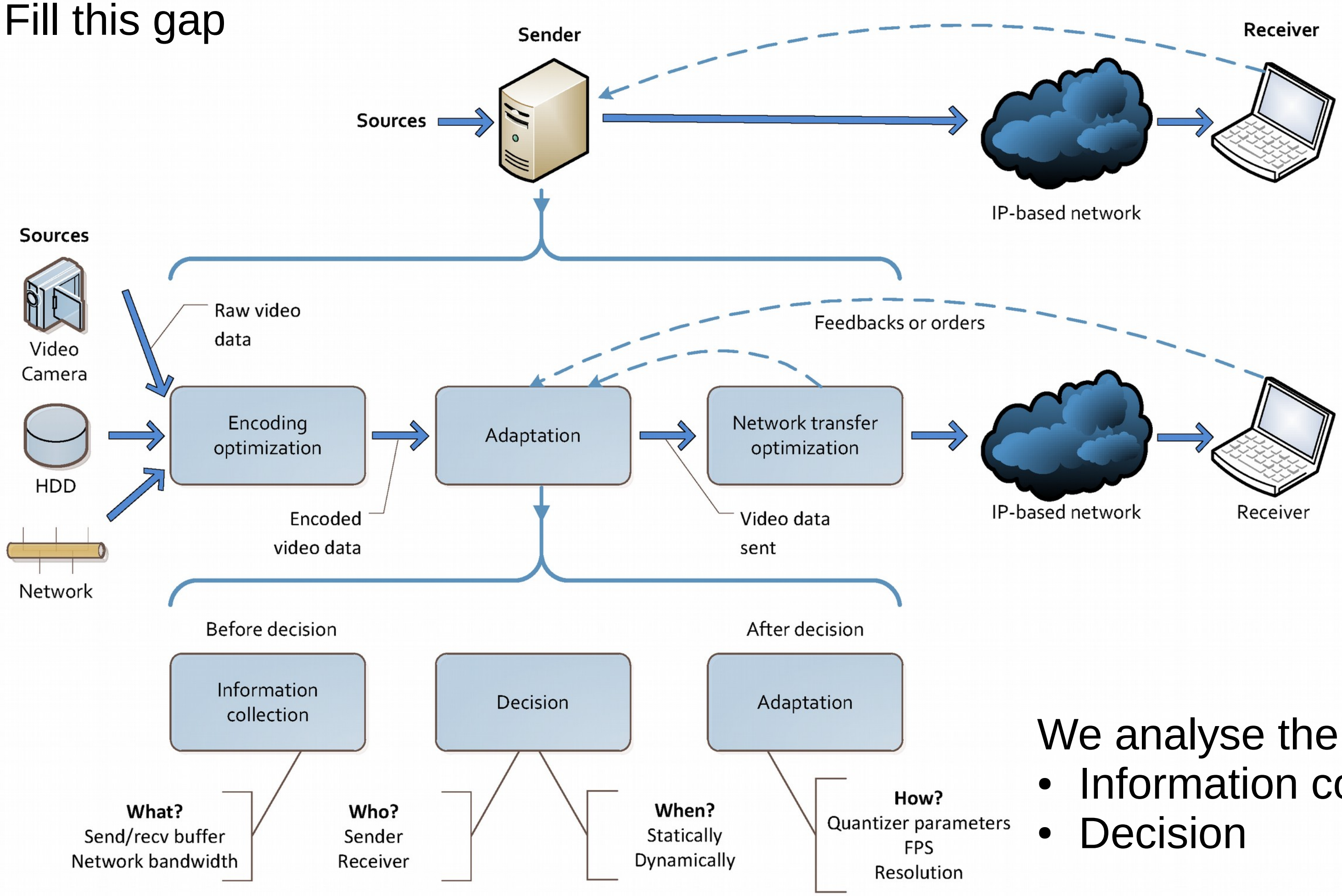
	ZVAAL	Q3	Q2	Q1	Q0
Sent	41418	68623	46818	24858	12938
Received	37683	40698	35531	24232	12828
Lost	3735	27925	11287	626	110

Conclusion: Our method has a much better trade-off sent/received/lost packets compared to static encoding

2.3 Adaptive video streaming with CC Taxonomy of adaptation params 1/3

W. Ramadan, PhD student

Reason: Many adaptation methods found in the literature, but no article classifying them
Goal: Fill this gap



We analyse the first two steps:

- Information collection
- Decision

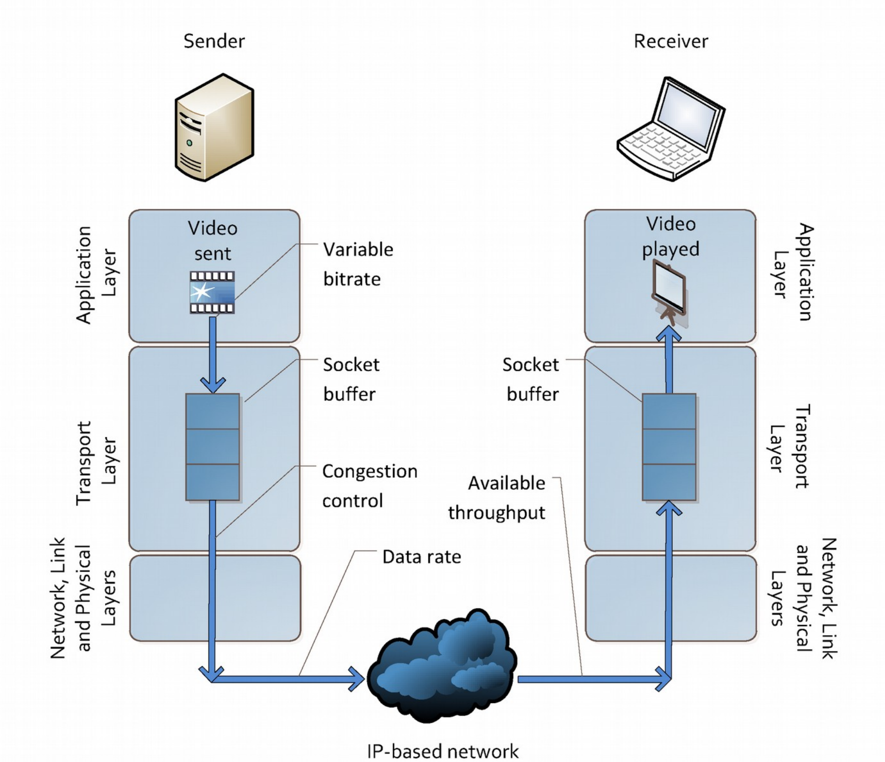
2.3 Adaptive video streaming with CC Taxonomy of adaptation params 2/3

W. Ramadan, PhD student

Why are there different adaptation methods?

Complexity of adaptive video transfer

Various speeds involved



Groups of adaptation methods:

- using information from sender buffer
- using information from receiver buffer
- using information from network
- hybrid
- using information from network, HTTP (proposed by major companies)

	Sender app	Network	Receiver app
Secs	const or dyn (ctrl)	dyn (non ctrl)	const
Bytes	dyn (ctrl)	dyn (non ctrl)	dyn (sender&net-ctrl)

2.3 Adaptive video streaming with CC Taxonomy of adaptation params 3/3

W. Ramadan, PhD student

	Context		Functioning			
	Beforehand data needed	Usage	Who	What	When	Interval
VAAL [159]	no	real-time	snd	snd buf, bytes	each n sec. (2 for ex.)	last RTT or 20 last pkts
QAC [52]	no	delay	snd	snd buf, bytes	when buf exceeds thresholds	instant
Buffer-driven [119]	no	real-time	snd	rcv buf	each RTCP interval	instant
MVCBF [125]	few	delay	snd	rcv buf, secs	when feedback arrives	whole interval
AHDVS [51]	few	delay	rcv	rcv buf, bytes	each 1.2 sec.	N/A
Kofler [114]	no	real-time	snd	net, bytes	when feedback arrives	RTT
Eberhard [65]	all	delay	snd	net, bytes	at RTSP feedback	N/A
Wien [191]	all	delay	snd	net, bytes	when feedback arrives	N/A
VTP [14]	all	delay	snd	net, bytes	each k pkts	many previous periods
Gorkemli [87, 89]	no	real-time	snd	net, bytes	when buf exceeds thresholds	a fixed interval
MPEG-TFRCP [188]	no	real-time	snd	net, bytes	each 32*RTT	whole interval
Evalvid-RA [121]	no	real-time	snd	net, bytes	each GOP	instant
CBVA [73]	all	delay	snd	net, secs	each GOP	instant
DRDOBS [194]	few	delay	snd	net, secs	when buf exceeds thresholds	instant
Schierl [172]	no	delay	snd	net, secs	each RTCP feedback	instant
RAAHS [126]	few	delay	rcv	net, secs	each segment received	whole interval
Nguyen [142]	no	real-time	snd	net, bytes & rcv buf	each GOP	instant
Major companies [204, 64, 145, 100]	all	delay	rcv	net, secs & rcv buf	each packet received	whole interval/instant

Conclusions:

- Major companies need beforehand data
- Generally, the adaptation decision is taken by sender, but major companies use receiver
- All values are used for what parameter: sender/receiver/network using bytes/seconds
- There is no consensus on interval parameter
- There are so many methods because there is no clearly best parameter

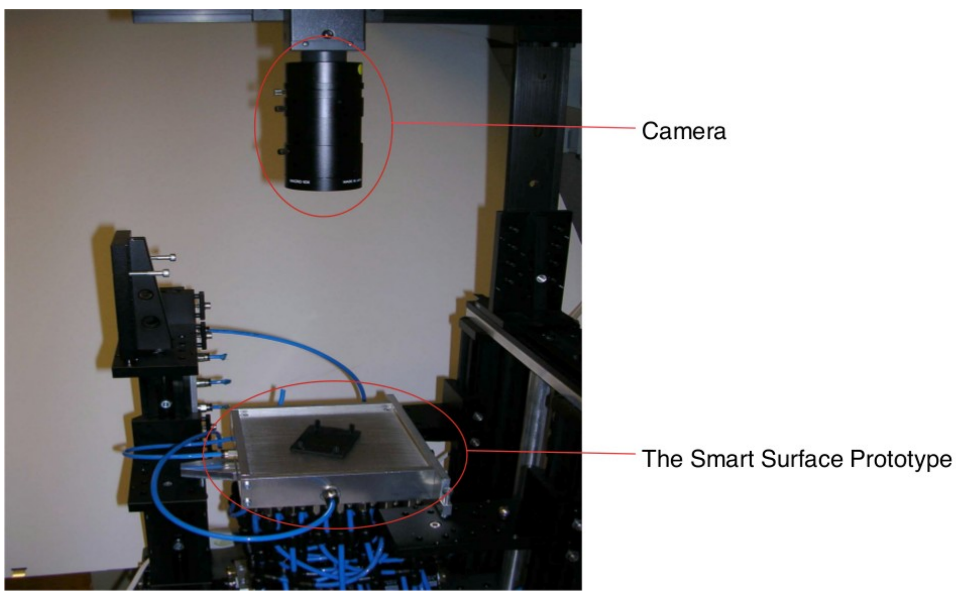
3.1 Communication in diMEMS Smart surface project

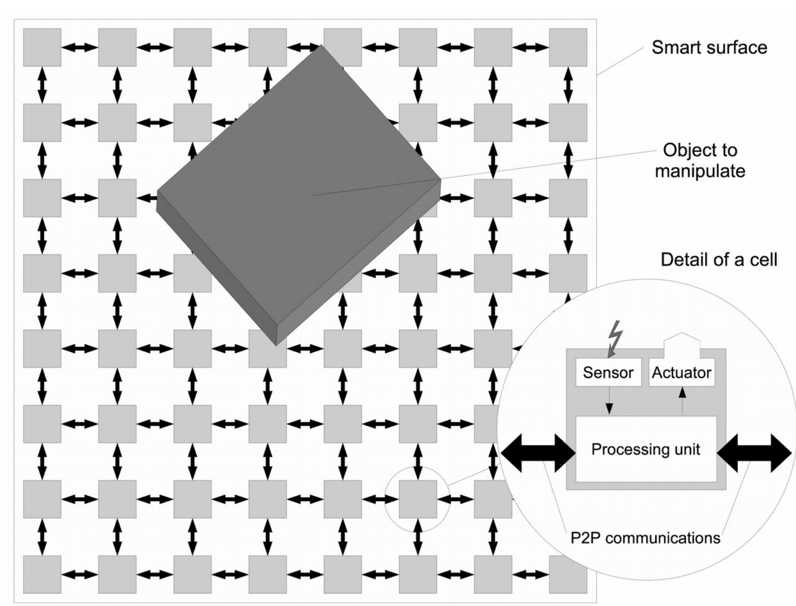
K. Boutoustous, PhD student

Goal: Design a distributed surface composed of numerous sensor/actuator cells for sorting and conveying micro objects/parts

Challenges:

- Recognise low resolution objects (e.g. 3x3)
- Multi-disciplinary project
- Should work in practice

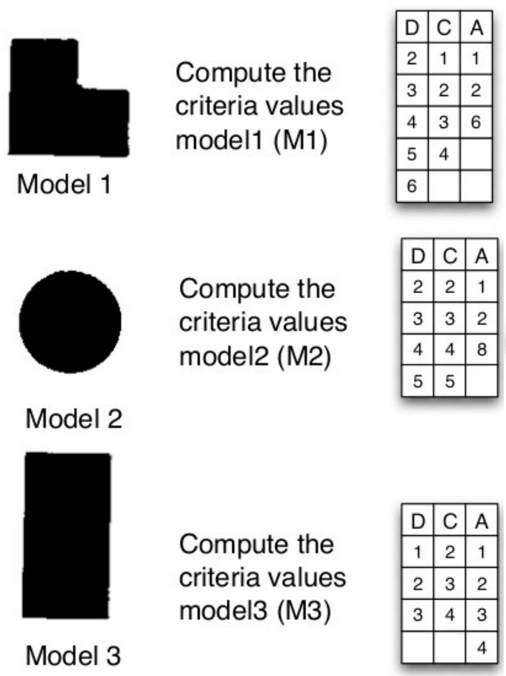




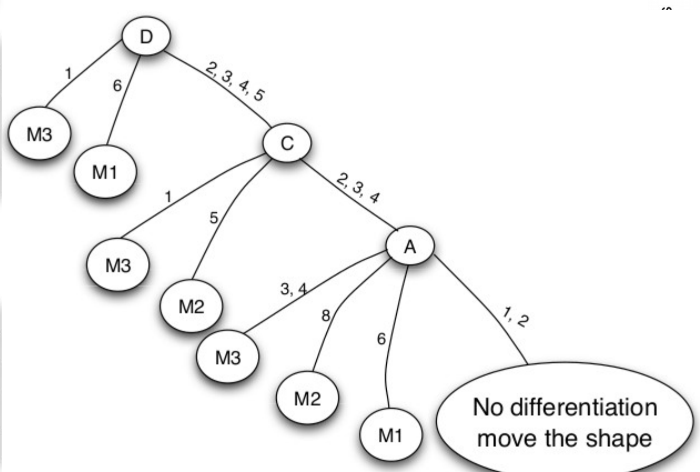
3.2 Communication in diMEMS
Find best surface size

K. Boutoustous, PhD student

Offline stage:
for each model
  for each rotation of 1°
    for each translation of size/10 px
      for each sensor grid to test
        discretise model
        for each criterion
          add criterion value to database

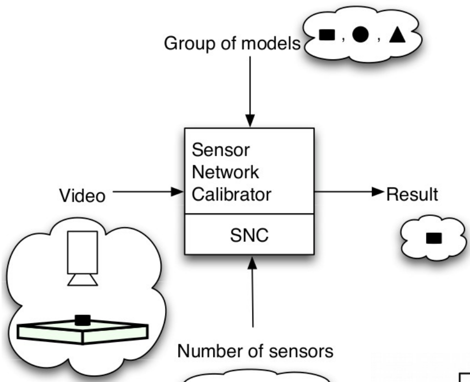


(a) Results of the offline phase



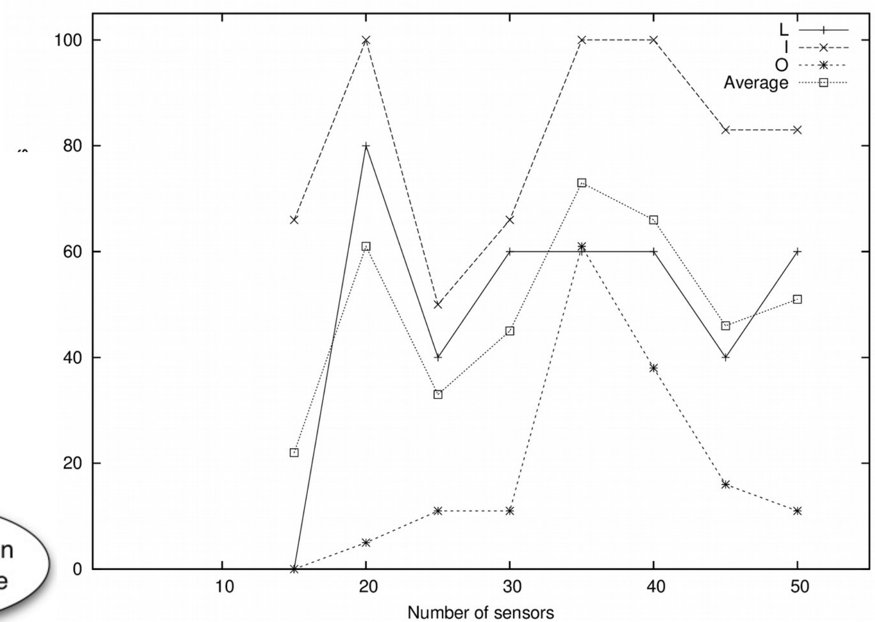
(b) Result tree generated

Free rotation of parts
Experimental results



Online stage:
for each image of the video
  for each sensor grid to test
    discretise image
    compute criterion values
    check if part can be differentiated

Experimental results:



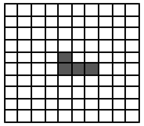
Conclusion: 35x35 yields best results

3.3 Communication in diMEMS Validation on functional platform 1/2

K. Boutoustous, PhD student

Offline stage, identical to previous slides

Online stage, uses distributed synchronous algorithms:



1. Reconstruction phase:

do surface_width + surface_height times
  communication step: each cell sends to its 4 neighbours its current view of the surface
  computation step: each cell merges its view with the 4 views received from neighbours
    => it increases by 1 cell its view of the surface
=> all cells obtain the same view of the object

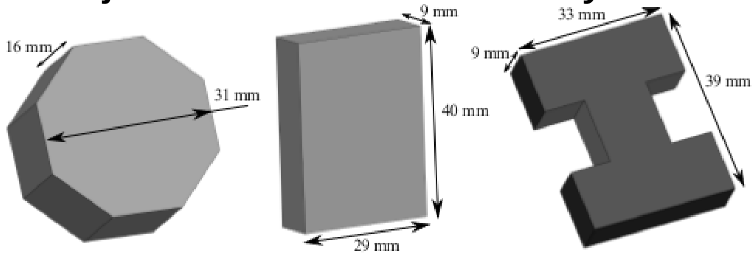
2. Differentiation phase:

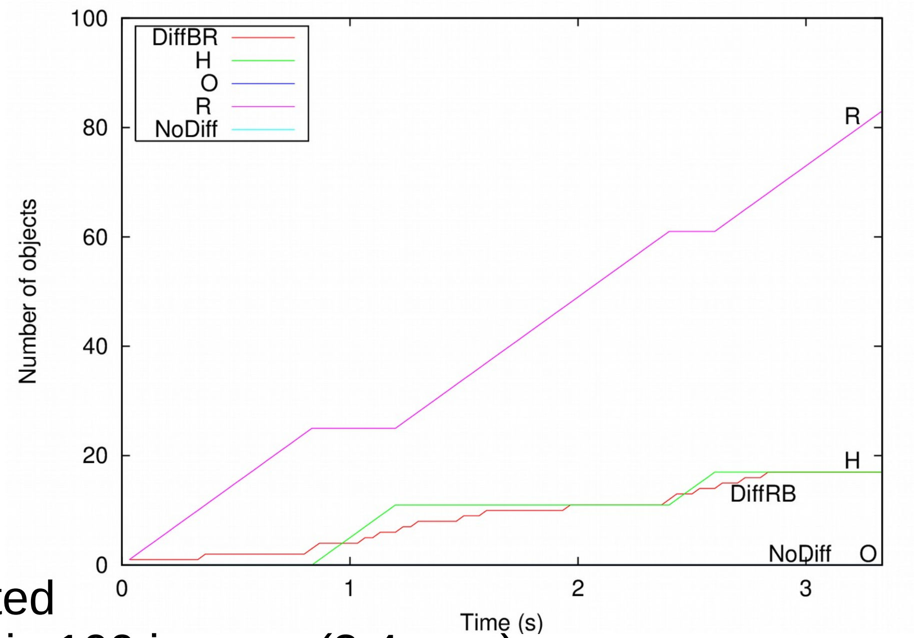
do
  each cell computes criterion values of the object
  each cell compares them with its database values
  if result is null
    object differentiated
  else
    move object
until object differentiated
inform control plane to move the object to the right destination

3.3 Communication in diMEMS Validation on functional platform 2/2

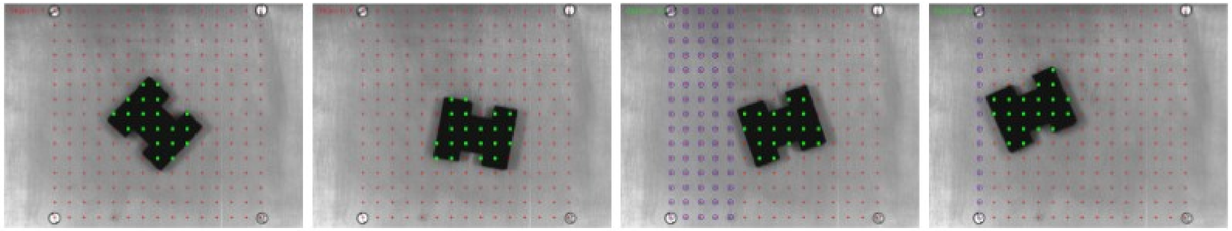
K. Boutoustous, PhD student

Objects to sort and convey:





In practice, objects can be unrecognised
or even wrongly differentiated
To cope with this, an object is considered differentiated
when it is recognised at least 60 times as one type in 100 images (3.4 sec.)



(a) t=0.4s, the object is not identified (b) t=2.4s, the object is still not identified (c) t=3.4s, the object is identified and the valves on its left are opening (d) t=4.1s, the object is moving toward the west edge of the surface

(show video ~/smart-surface/Boutoustous*.avi if have time)

3.4 Communication in diMEMS Enhanced part differentiation

K. Boutoustous, PhD student

With a single reference position:

$$g(j) = \frac{1}{q}\sum_{i=1}^{q}\left|\frac{r_i(j)}{c_i} - 1\right|, \quad j \in 1, \ldots, m$$

With several reference positions:

$$g'(j) = \min_{d \in D}\left\{\frac{1}{q}\sum_{i=1}^{q}\left|\frac{r_i^d(j)}{c_i} - 1\right|\right\}, \quad j \in 1, \ldots, m$$

q, number of criteria
m, number of models
$r_i(j)$, $r_i d(j)$, reference value(s) of criterion i on model j in database
ci, value of criterion on surface

The model the closest to 0 is considered

Simulation results on Sq, I, L parts when Sq part is on the surface:
Previous method:

Criterion	Sq (%)	I (%)	L (%)	Average
S	37.0	52.0	57.3	48.7
A	33.5	40.5	48.5	40.8
AS	59.5	74.0	68.0	67.1

Gap with single reference:

Criterion	Sq (%)	I (%)	L (%)	Average
S	100	99.0	98.0	99.0
A	100	99.0	78.0	92.3
AS	100	100	96.5	98.8

Gap with several references:

Criterion	Sq (%)	I (%)	L (%)	Average
S	100	99.0	98.0	99.0
A	100	99.0	78.0	92.3
AS	100	99.5	79.0	92.8

Conclusions for methods with gaps:
- Recognise parts better when using a single image of the part
- Particularly useful when cells are faulty or objects are altered/deformed

4.1 Communication in nanonetworks
Nanonetwork Minum Energy coding

M. A. Zainuddin, PhD student

Context: in TS-OOK modulation, sending bit 1 consumes energy, whereas bit 0 does not, since it is simply not sent
Goal: reduce energy consumption by replacing in data to be sent bits 1 by bits 0 as much as possible
Idea: encode more often used symbols with fewer 1s, similar to Huffman algorithm

Algorithm:

Bits to be sent:		Dict:		Bits actually sent:
11 10 00 11 10 01 11	->	11 3 00	->	00 01 10 00 01 11 00
(9 bits 1)		10 2 01		(5 bits 1)
		00 1 10		=> 45% energy reduction
		01 1 11		

Properties:

- up to 100% energy reduction (11..11 -> 00..00)
- reduction greatly depends on input data, e.g.:
  - no reduction for highly compressed files (mp4 and jpg)
  - 20–40% reduction for uncompressed files (bmp, yuv and dll)
- the greater the symbol length, the greater the reduction, but the greater the dictionary
- sensible to data transmission errors: 1-bit error during transmission leads to 4-bit error

Conclusions and perspectives

- I have been working on four fields, all related to optimisation of network communication
- I have been using simulations, experiments, numerical results, and formalisation to validate my ideas
- Most of my articles present new ideas, but 1–2 of them are analysis articles

- Nanonetworks will develop, and their peculiarities need to be taken into account
- Tb/s communication is promising
  - Edholm's law of bandwidth (Eslambochi): *"Wireless data rates have doubled every 18 months over the last three decades"*
  - J. Jornet: *"I have always been taught that communication is more expensive than computation, but this will no longer be true"* => new communication models will be needed

We live in the age of communication, witnessed by online social networking, videoconferencing, Internet of objects... Network communication has a bright future!

Additional slides

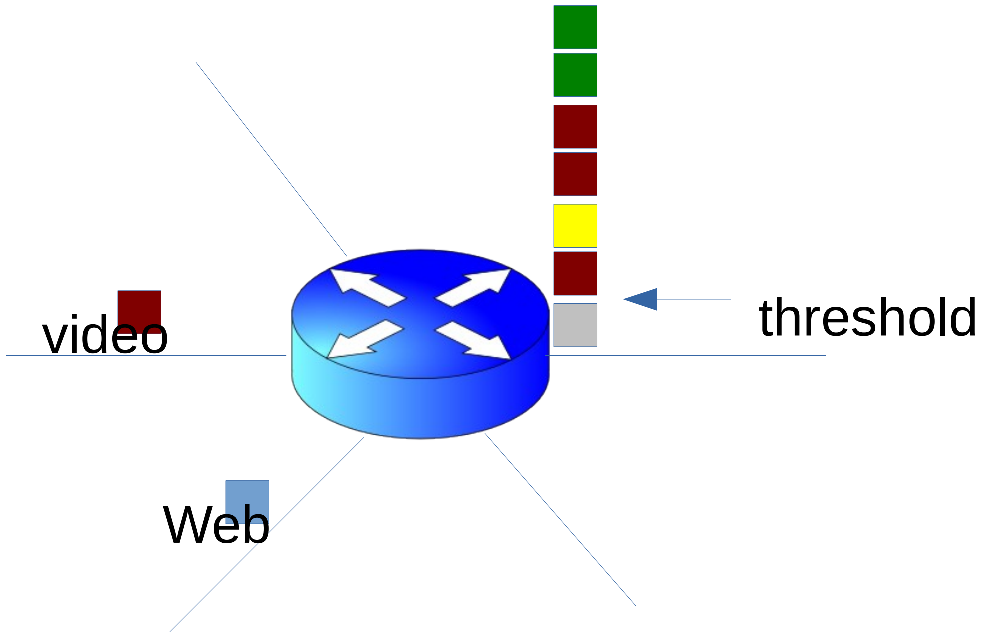
1.0 Congestion control in networks FavourTail 1/2

No student involved

Problem: in current ultra-fast networks, Web pages, even if short, still take time to be downloaded

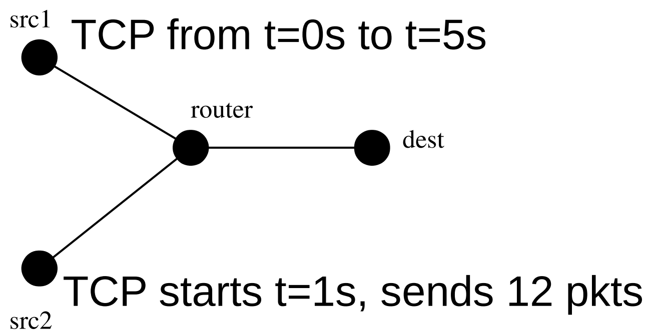
Goal: prioritise short flows (in detriment of long flows)

Idea: router has a pointer dividing the queue in two: favoured packets and normal ones; when a packet needs to be inserted in a router queue, it is added to favoured queue iff no other packet of the same flow exists in the queue

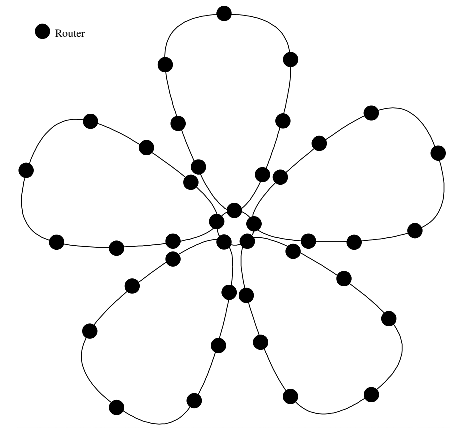


1.0 Congestion control in networks FavourTail 2/2

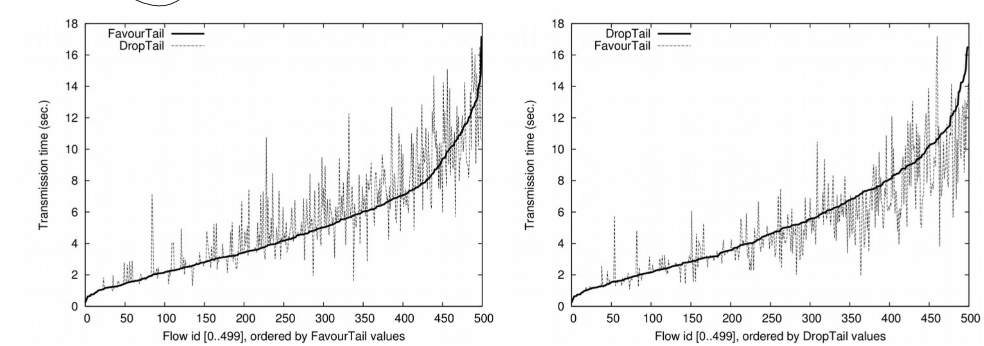
No student involved



Router is (1) DropTail, (2) FavourTail
1st flow sends 591 packets in both cases
2nd flow, trtime = 0.53s for DropTail, 0.43s for FavourTail => 20% gain
Analysis: 1st packet overtakes 13 packets, the 2nd one 14 packets, all the others are not prioritised



All routers are
(1) DropTail, (2) FavourTail
500 TCP flows with random src/dest sending random 10–600 packets



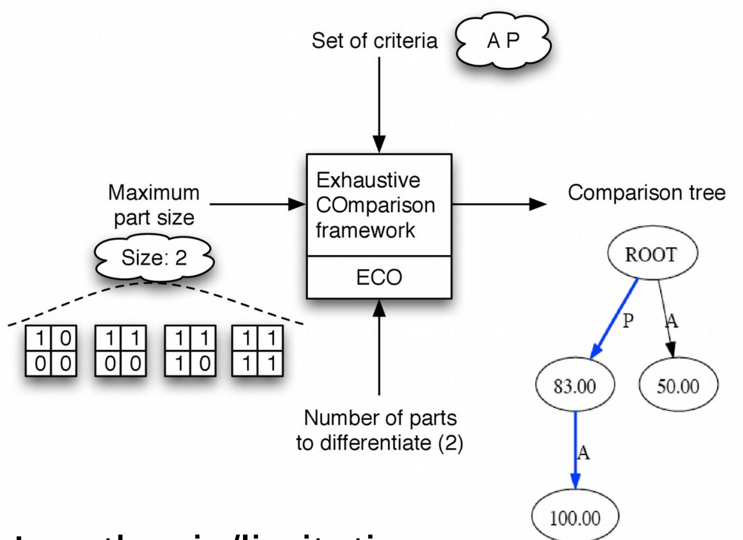
	DropTail	FavourTail
Tr. time	2618	2410
Lost pkts	2470	1608

Conclusions:

- **Intuitively**, short flows are favoured
- **Surprisingly**, all the flows are generally favoured
- So **global** metrics get better

3.2 Communication in diMEMS Find best criteria

K. Boutoustous, PhD student



Hypothesis/limitation:
No rotation for parts

Part generation:

3x3 -> 2^9 = 512 parts -> 35 unique parts

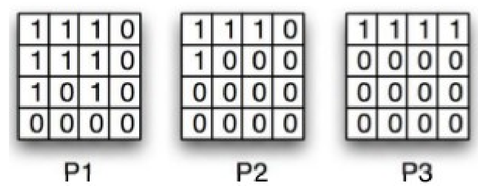
$C_{35}3$ = 6545 groups

4x4 -> 2^16 = 65536 parts -> 1280 unique parts

$C_{1280}3$ = 348 millions groups

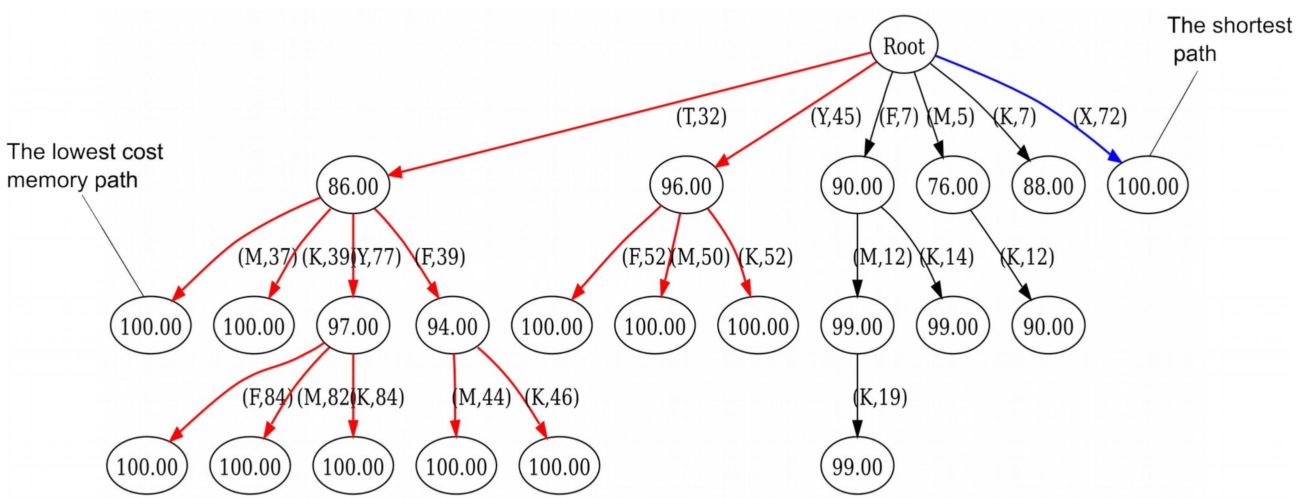
Differentiation percentage computing for three parts:

Let G1 a group of parts:



	S	P	A
P1	8	14	6
P2	4	10	4
P3	4	10	5

Value of associated criterion for each part



3.3 Communication in diMEMS Find best surface size 2/2

K. Boutoustous, PhD student

Why non differentiation percentage (NDR) is NOT a decreasing function of grid size?
Not due to quantisation effects per se it seems (because a big line and a small square are always differentiated)
Possible explanations:

- results depend on models; hypothetical counter-example, showing values of one criterion for two models for 15x15 and 20x20 grid sizes (likely)
- results depend on video images, which show only SOME positions of parts (less likely)