Deployment of mission-critical surveillance applications on wireless sensor networks

NETCOM research group, IMDEA
University Carlos III of Madrid
Wednesday, May 19th, 2010

Prof. Congduc Pham
http://www.univ-pau.fr/~cpham
Université de Pau, France

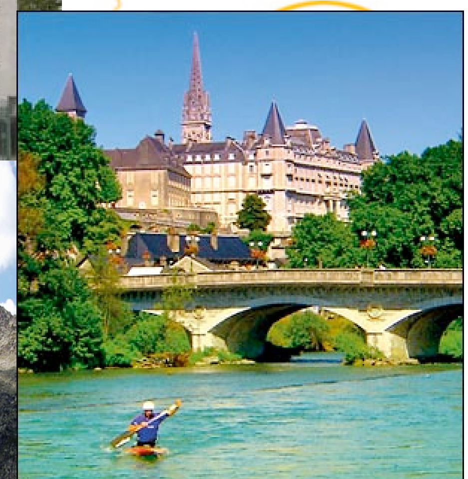
SITY OF PAU

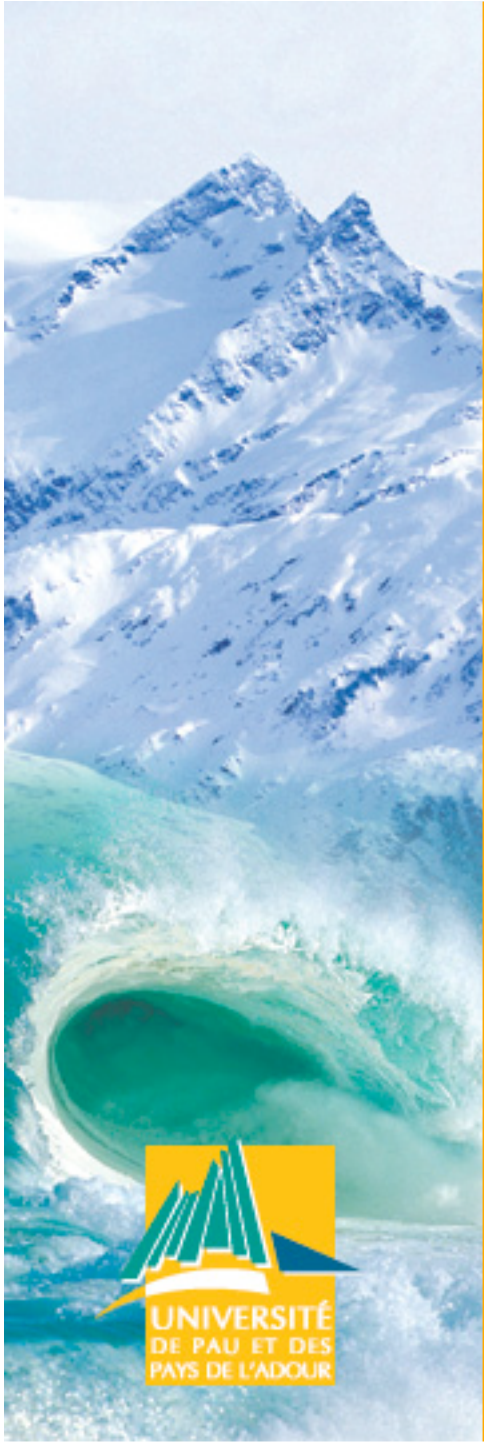
SES

Diaporama des
Campus de l'UPPA

THE 3 GE
SITES OF

ayonn
nglet





LIUPPA

COMPUTER SCIENCE LAB

- 32 FACULTY MEMBERS
- 25 PHD STUDENTS

2 RESEARCH TEAMS

- MODELING, VISUALIZATION, EXECUTION & SIMULATION
- INFORMATION PROCESSING, INTERACTIONS AND ADAPTATION

Deployment of mission-critical surveillance applications on wireless sensor networks

Deployment of mission-critical surveillance applications on wireless sensor networks

Surveillance applications (1)

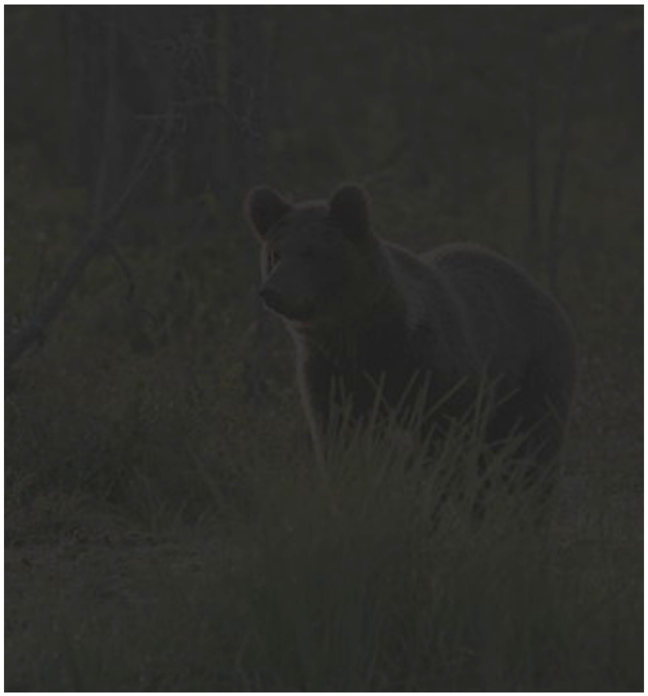
- Lesson 1: don't miss important events

Whole understanding of the scene is wrong!!!

What is captured

Surveillance applications (2)

- Lesson 2: high-quality not necessarily good



333x358 16M colors, no light

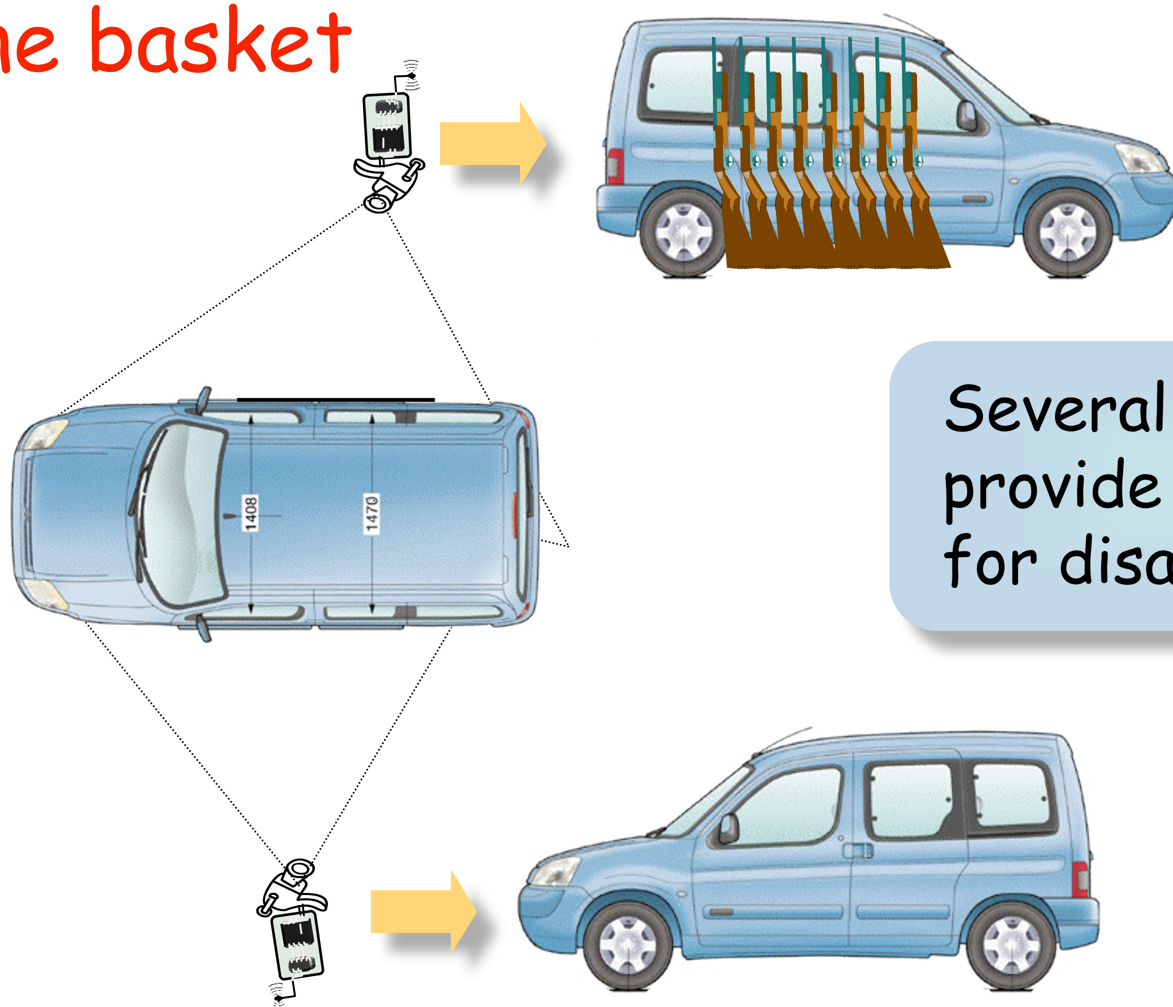
167x180 16 colors, light

Keep in mind the goal of the application!

167x180 BW (2 colors), light

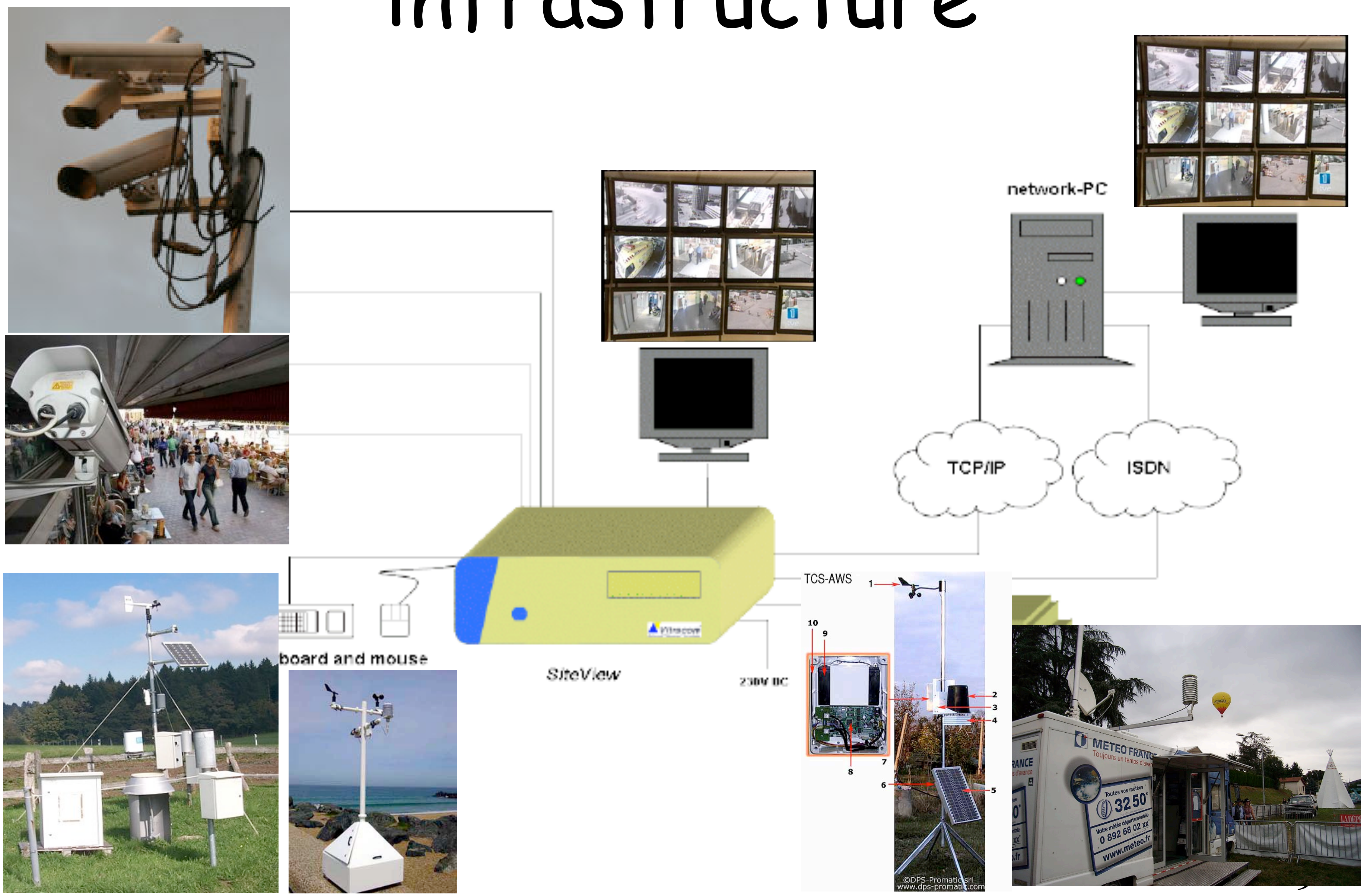
Surveillance applications (3)

- Lesson 3: don't put all your eggs in one basket



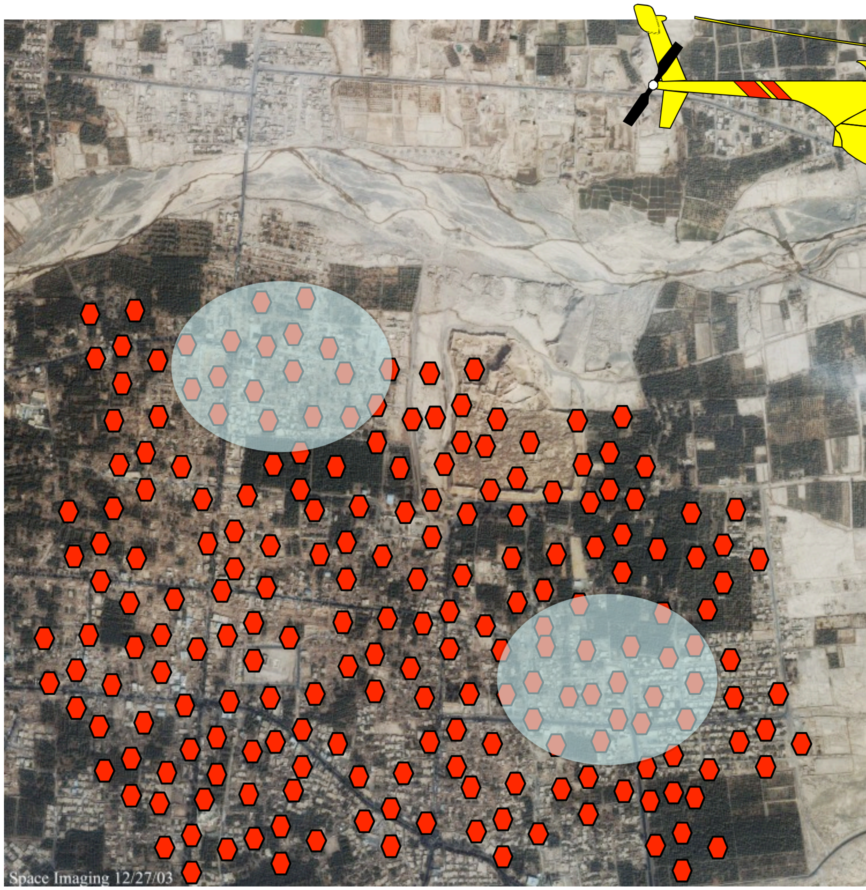
Several sources provide multi-view for disambiguation

Traditionnal surveillance infrastructure



Small, Automous Sensors

disaster relief - security

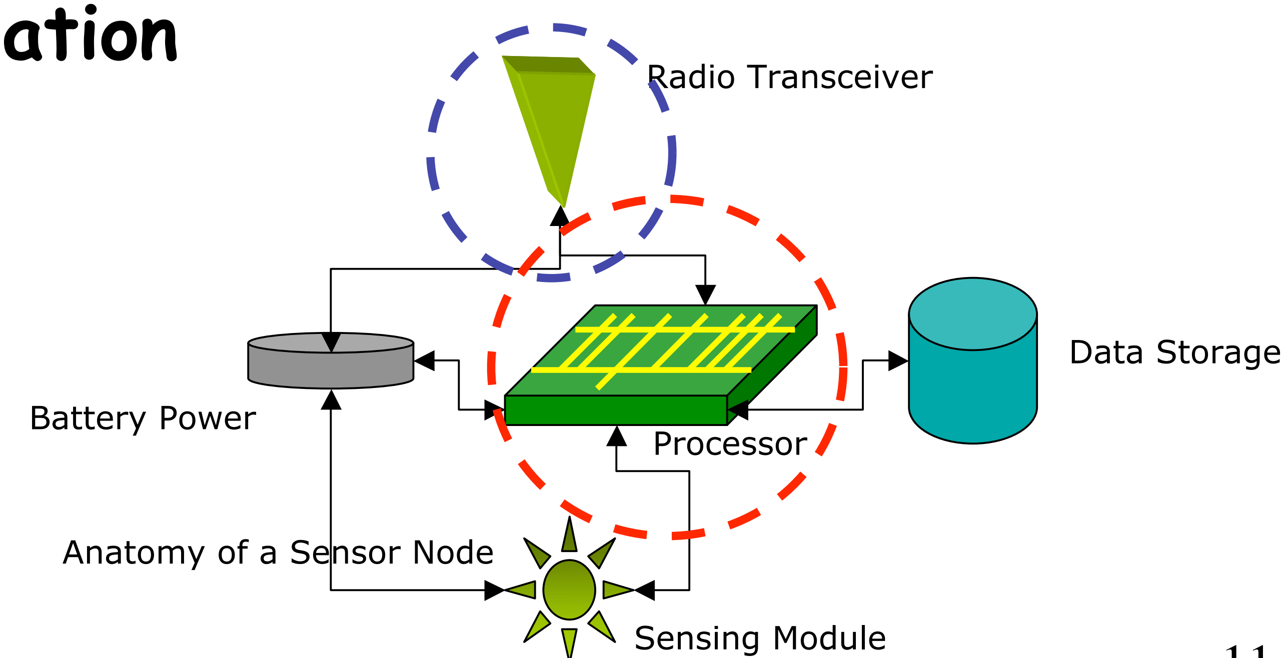


Organization of rescue in large scale disasters relief operations

Rapid deployment of fire detection systems in high-risk places

Wireless autonomous sensor

- In general: low cost, low power (the battery may not be replaceable), small size, prone to failure, possibly disposable
- Role: sensing, **data processing**, **communication**



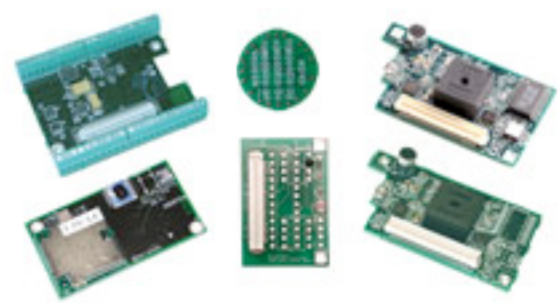
Berkeley Motes (contd.)

- Each Mote has two separate boards
  - A main CPU board with radio communication circuitry
  - A secondary board with sensing circuitry
- Decouples sensing hardware from communication hardware
- Allows for customization since application specific sensor hardware can be plugged-on to the main board


MICA2

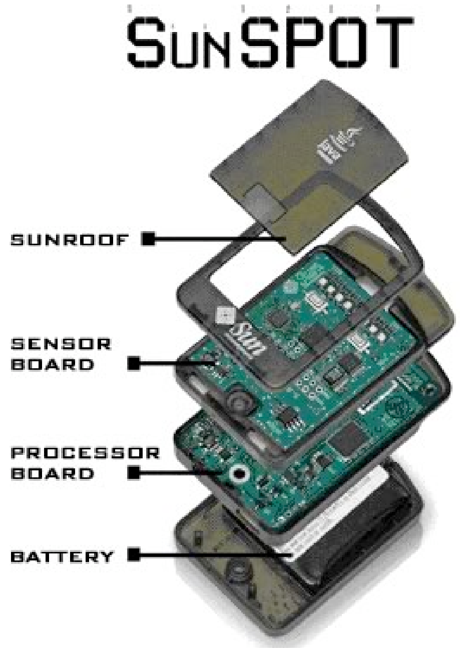

Imote2

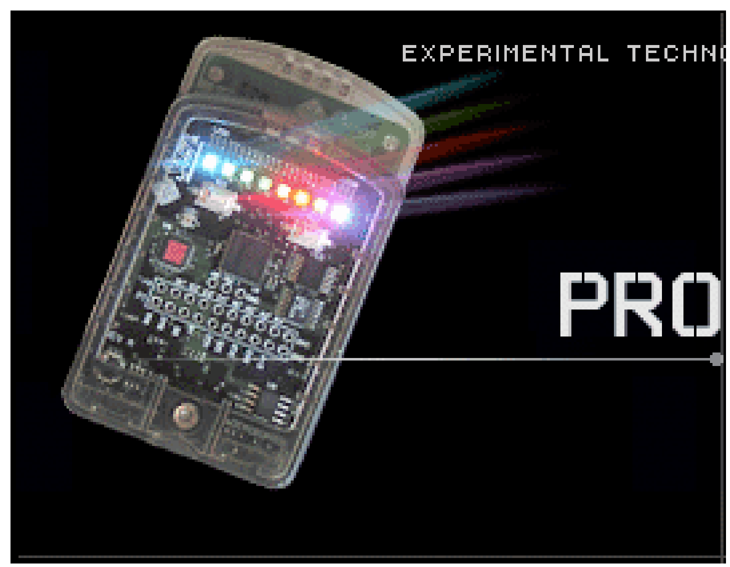

MICAz


Sensing boards

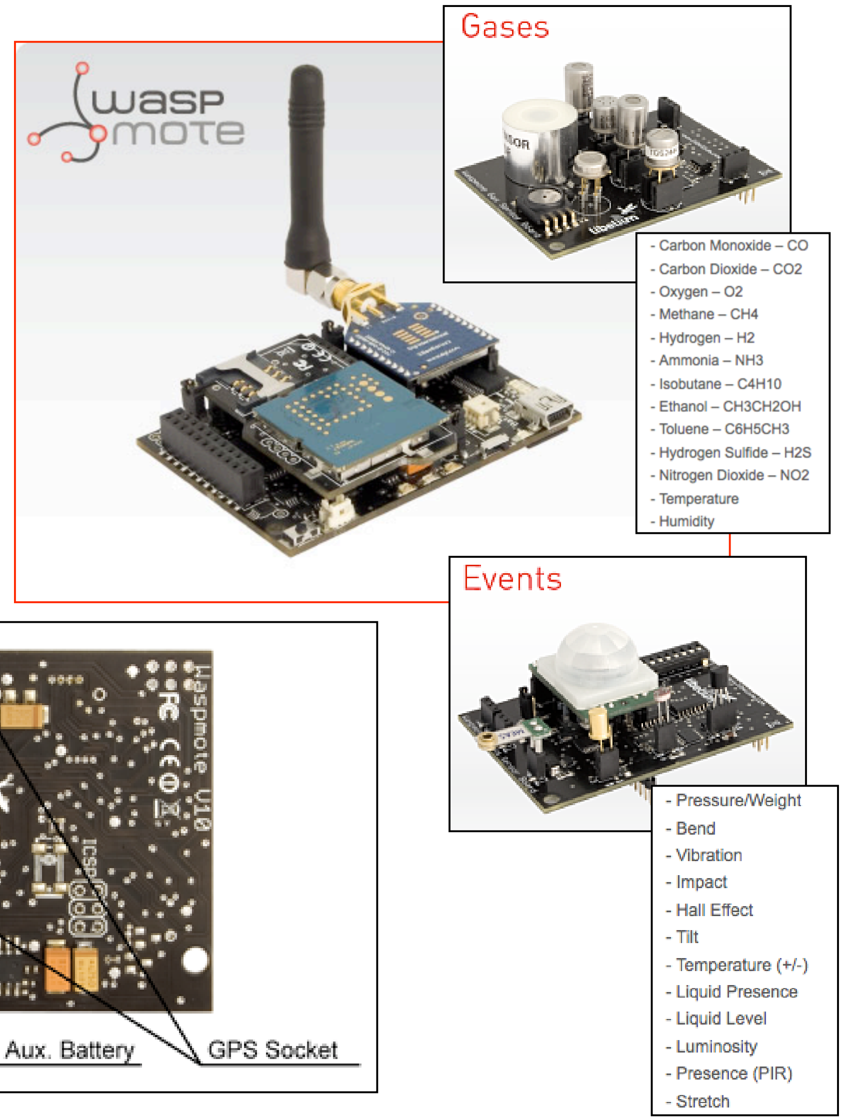
SUN SPOT

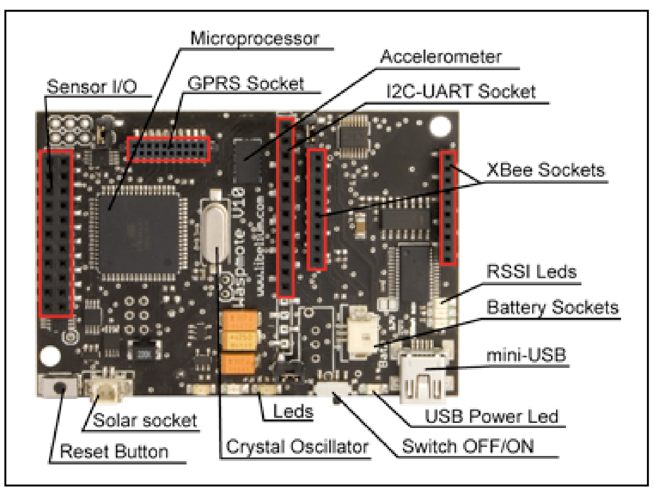
- Processor : ARM920T 180MHz 32-bit
- 512K RAM & 4M Flash.
- Communication : 2.4GHz, radio chipset: TI CC2420 (ChipCon) – IEEE 802.15.4 compatible
- Java Virtual Machine (Squawk)
- LIUPPA is official partner

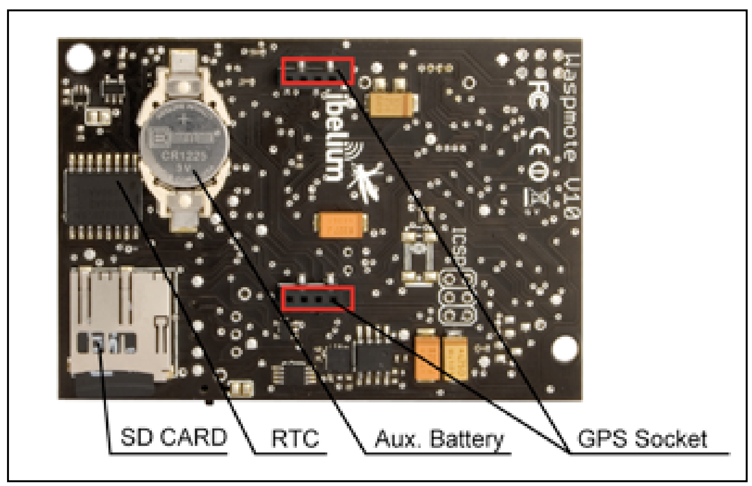




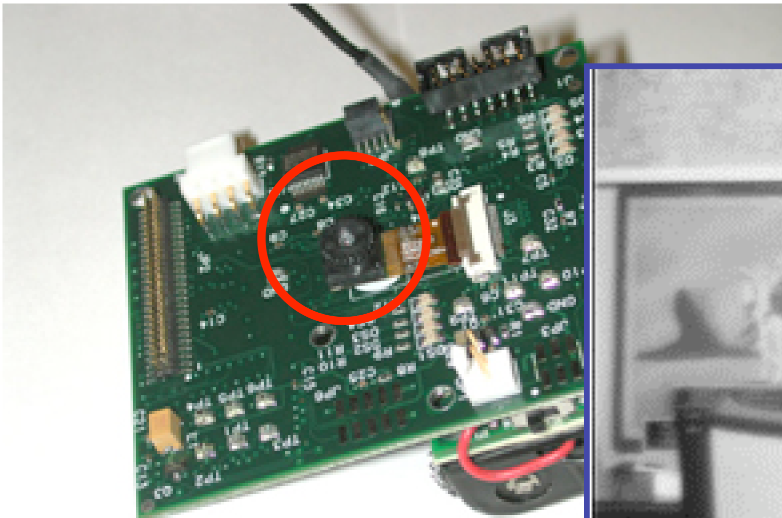
- ATmega1281 microcontroller
- 8K RAM & 1G SD card.
- 2.4GHz IEEE 802.15.4 compatible. RF and GSM/GPRS



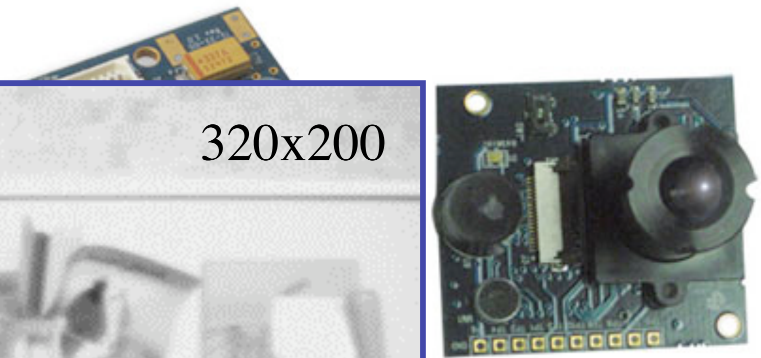




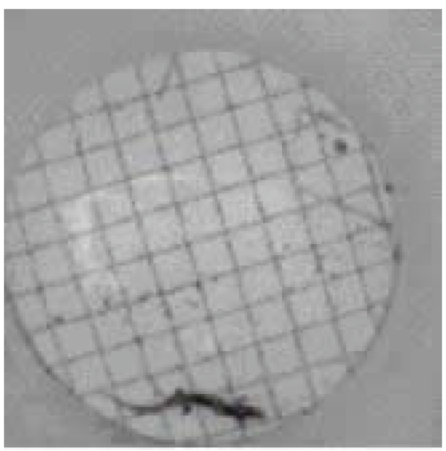
Wireless Video Sensors



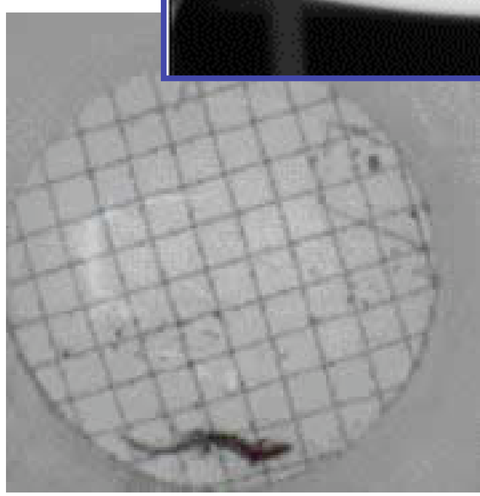
Cyclops video board on M



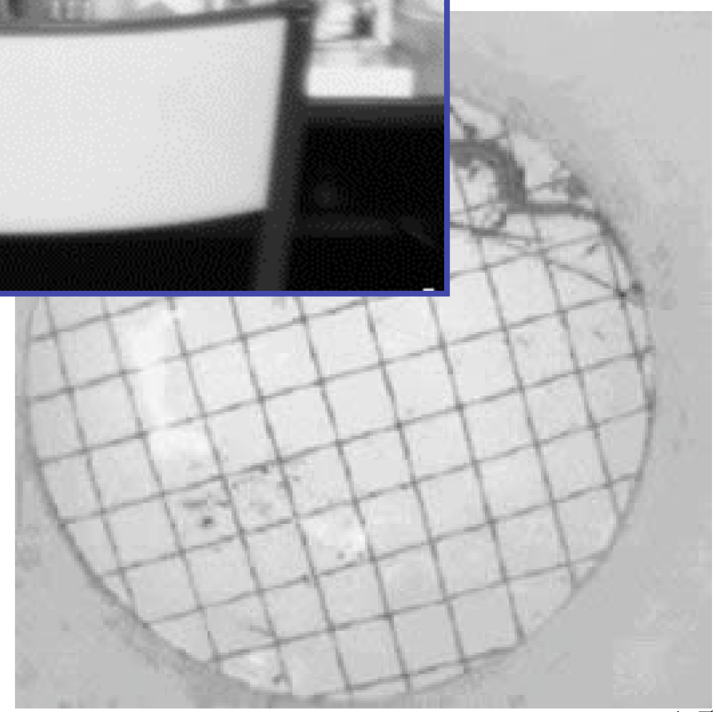
ultimedia board



128x128



140x140



240x240

Sensor as common object
toward very large scale deployment

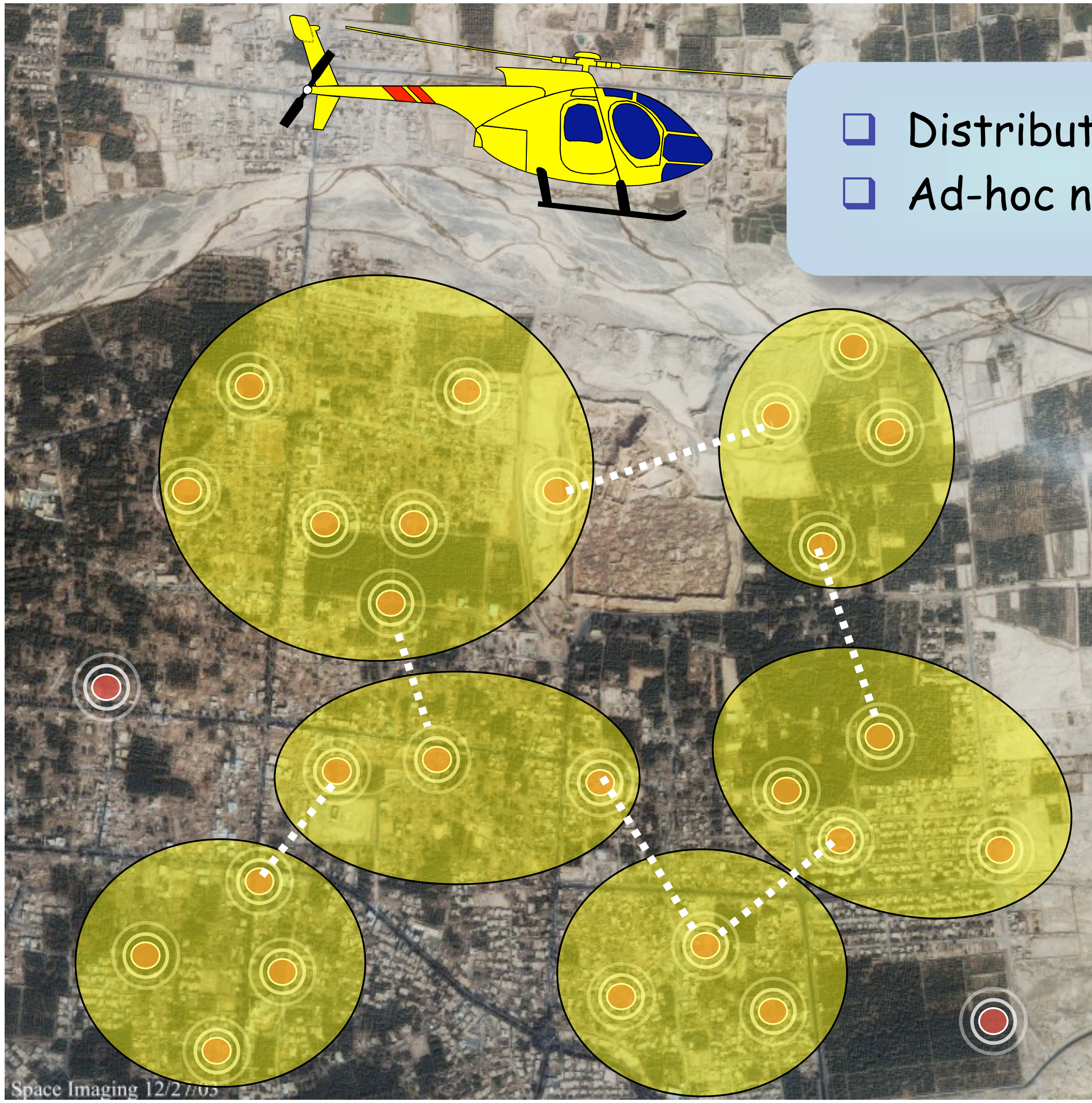
Environmental monitoring
- air
- water

Cell-phones with embedded CO sensor

- most ubiquitous device (millions)
- not deployment cost
- high replacement rate
- no energy constraints

Wireless Sensor Network (1)

Wireless Sensor Network (2)



- Distributed system
- Ad-hoc network

Mission-critical surveillance applications

- Availability: 24/24 surveillance
  - Hardware : failures, energy depletion
  - Data
- Quality
  - Enhance/validate/disambiguate information with several sources of information
  - Adaptation to local conditions
- Reliability/integrity
  - Hardware
  - Network
  - Data

QoS !

WSN

Internet

Ad-hoc networks

Embedded systems

multimédia

Surveillance as a Service

ACCOUNTABILITY

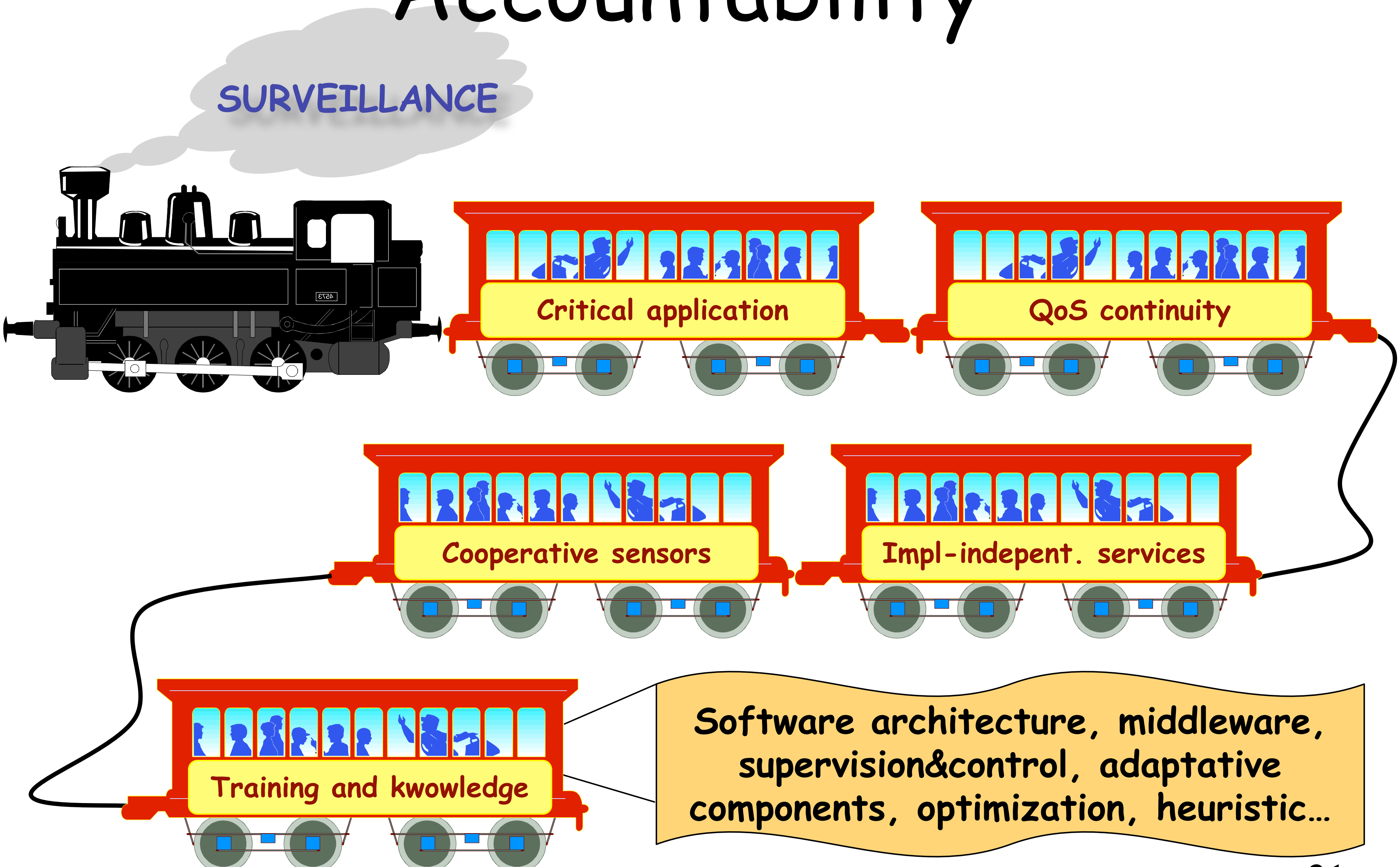
- Similar to Service Level Agreement

➔ SURVEILLANCE AT ANY PRICE ⬅

no discontinuity of service against node's failures

data availability and reliability

Surveillance as a service: Accountability

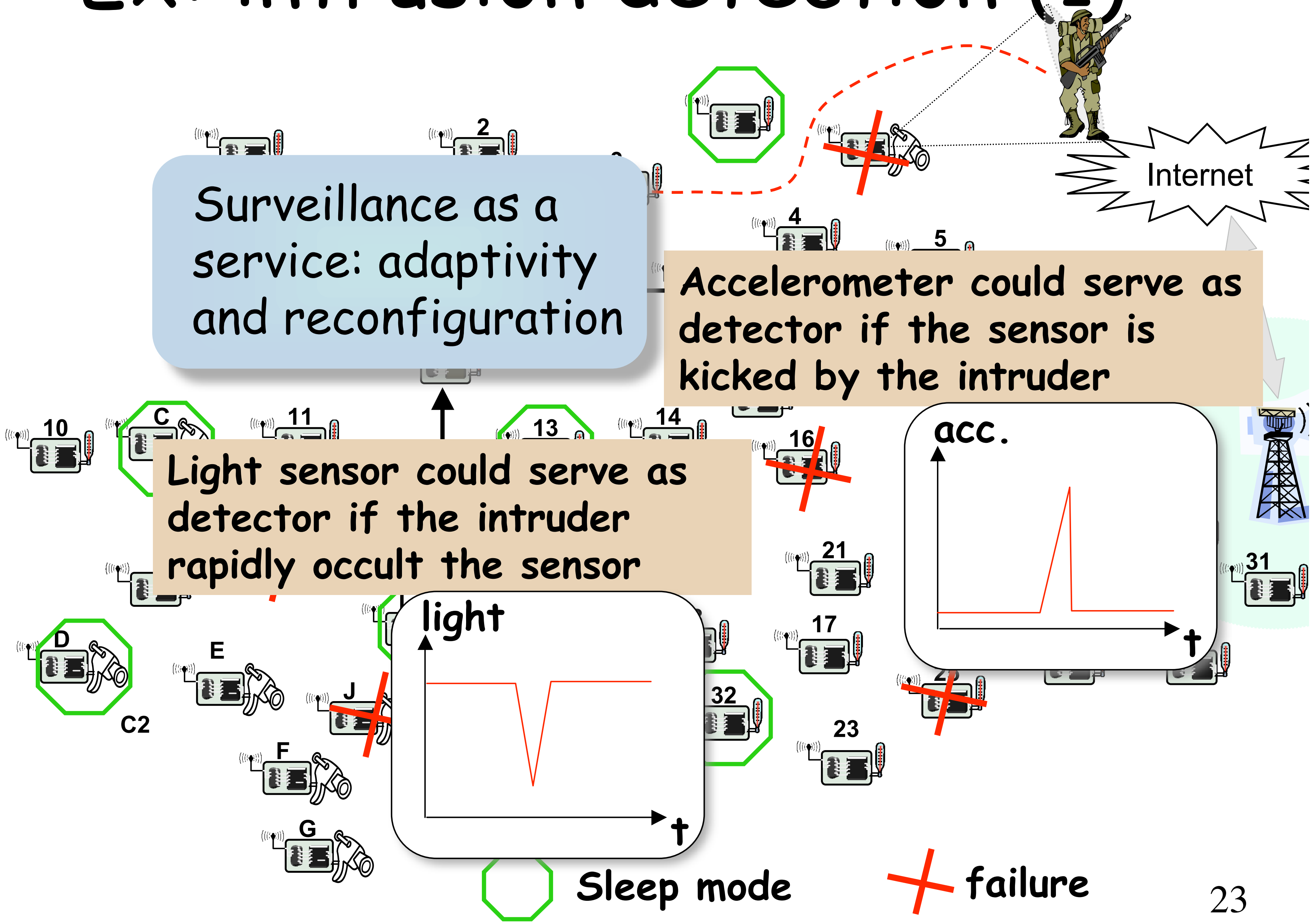


Ex: intrusion detection (1)

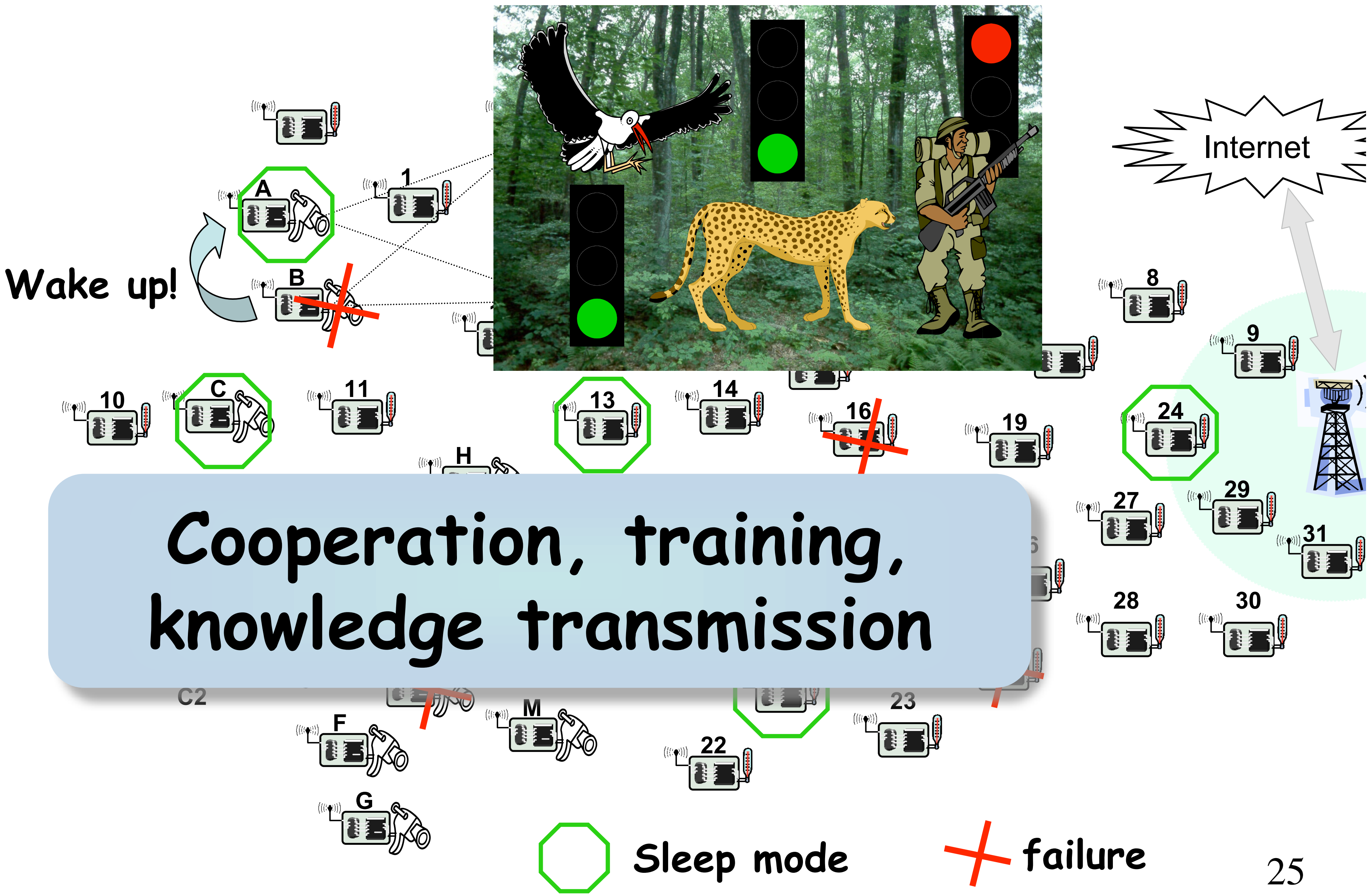
Surveillance as a service: adaptivity and reconfiguration

Accelerometer could serve as detector if the sensor is kicked by the intruder

Light sensor could serve as detector if the intruder rapidly occult the sensor



Ex: intrusion detection (2)



Cooperation, training, knowledge transmission

Towards smart sensor grids

- The ultimate goal is to define a customizable sensor grid that could be dynamically programmed according to the application's objectives and needs
- Similar to the so-called active networking concept for the Internet…
- …but much easier to achieve!!

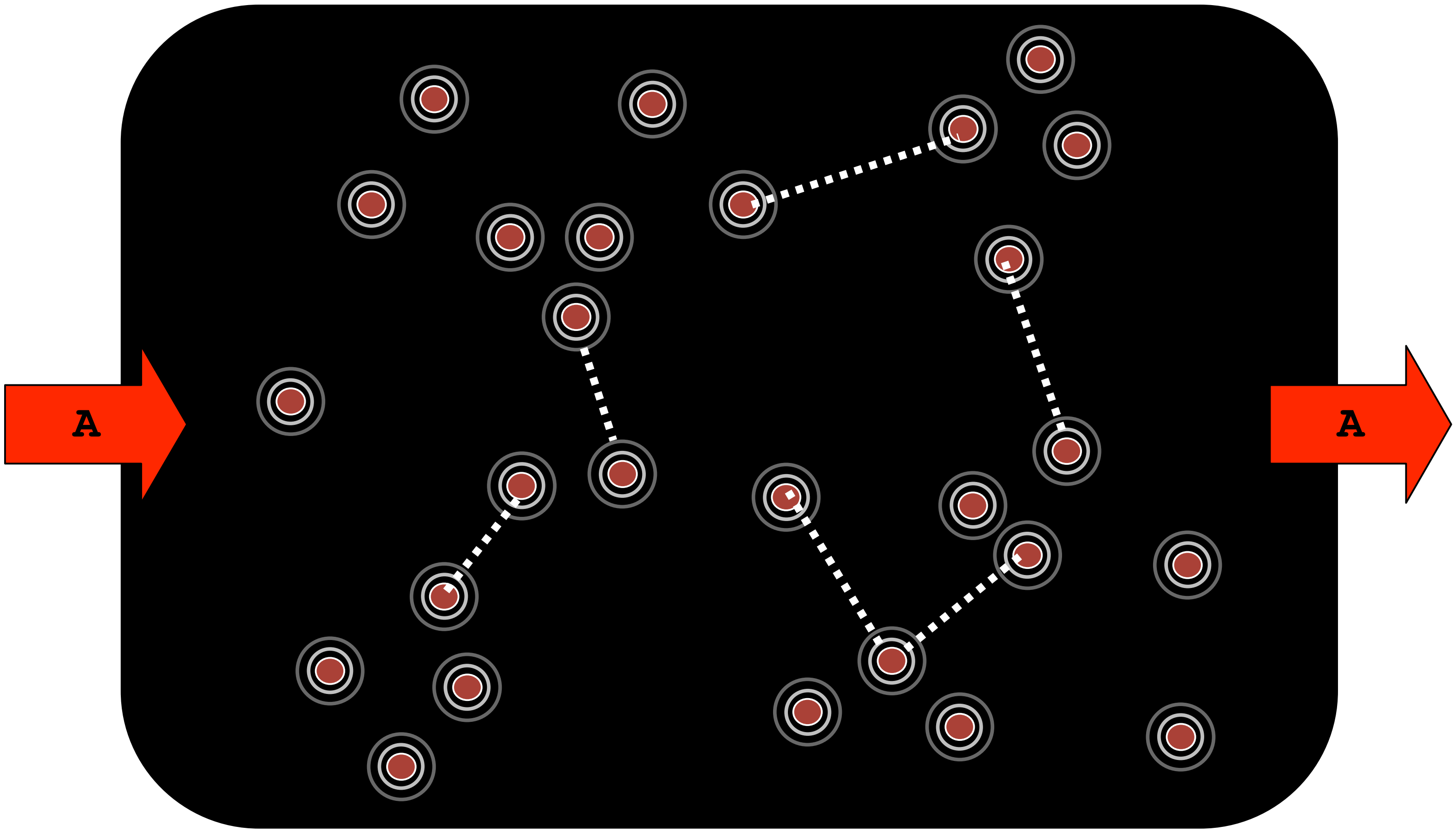
The Internet we wanted, the one we have...

End-to-end Quality of Service	No Quality of Service
Distributed processing	Very centralized architecture
Internet-wide, uniform control and policies	Heterogeneous and domain/ISP specific policies
Fast integration of new needs, new applications, new technologies	Upgrades and incremental deployments are slow

Net Neutrality or Not?

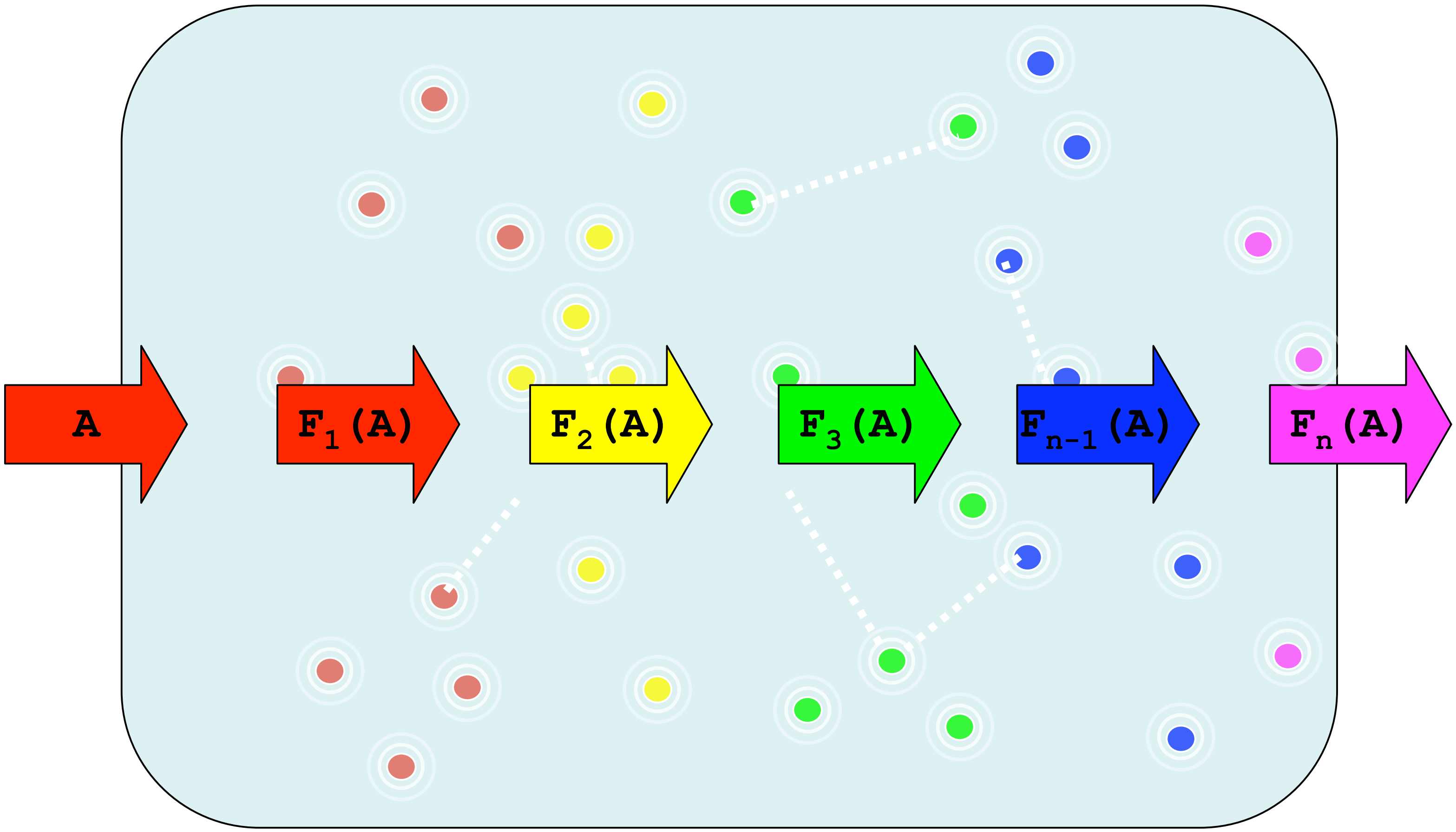
- The Internet's success is in a large part debtful to what's called Net Neutrality (IP neutrality)
- But, Net Neutrality is the main brake for achieving large scale QoS : IP routers only forward packets!
- Some services can be best supported or enhanced using information that is only available inside the network!
- Fortunately, in a sensor network, each node has de-facto specific processing capabilities

One vision for enabling QoS in Sensor Nets (1)



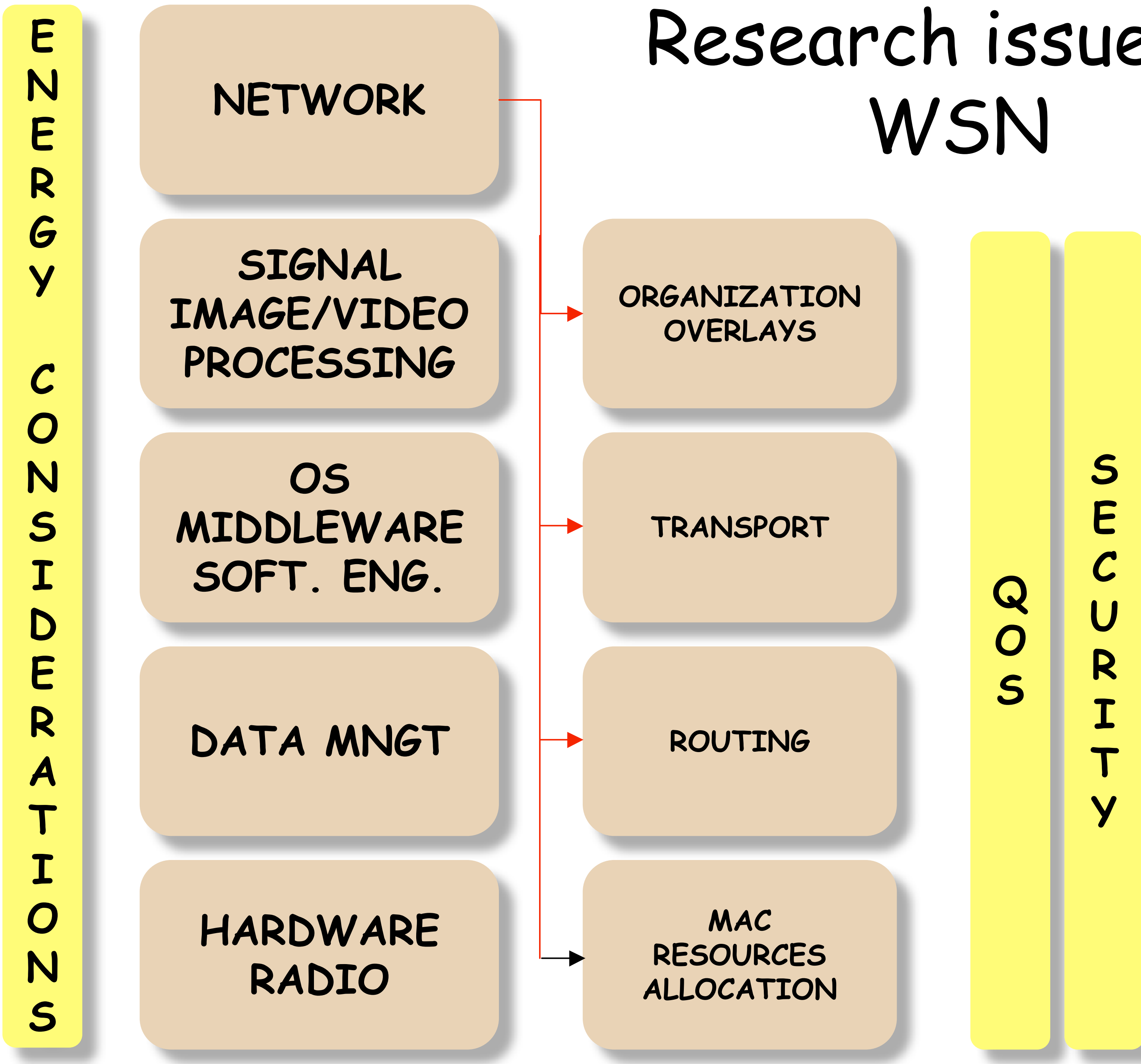
AVOIDS THE BLACK-BOX VISION

One vision for enabling QoS in Sensor Nets (2)

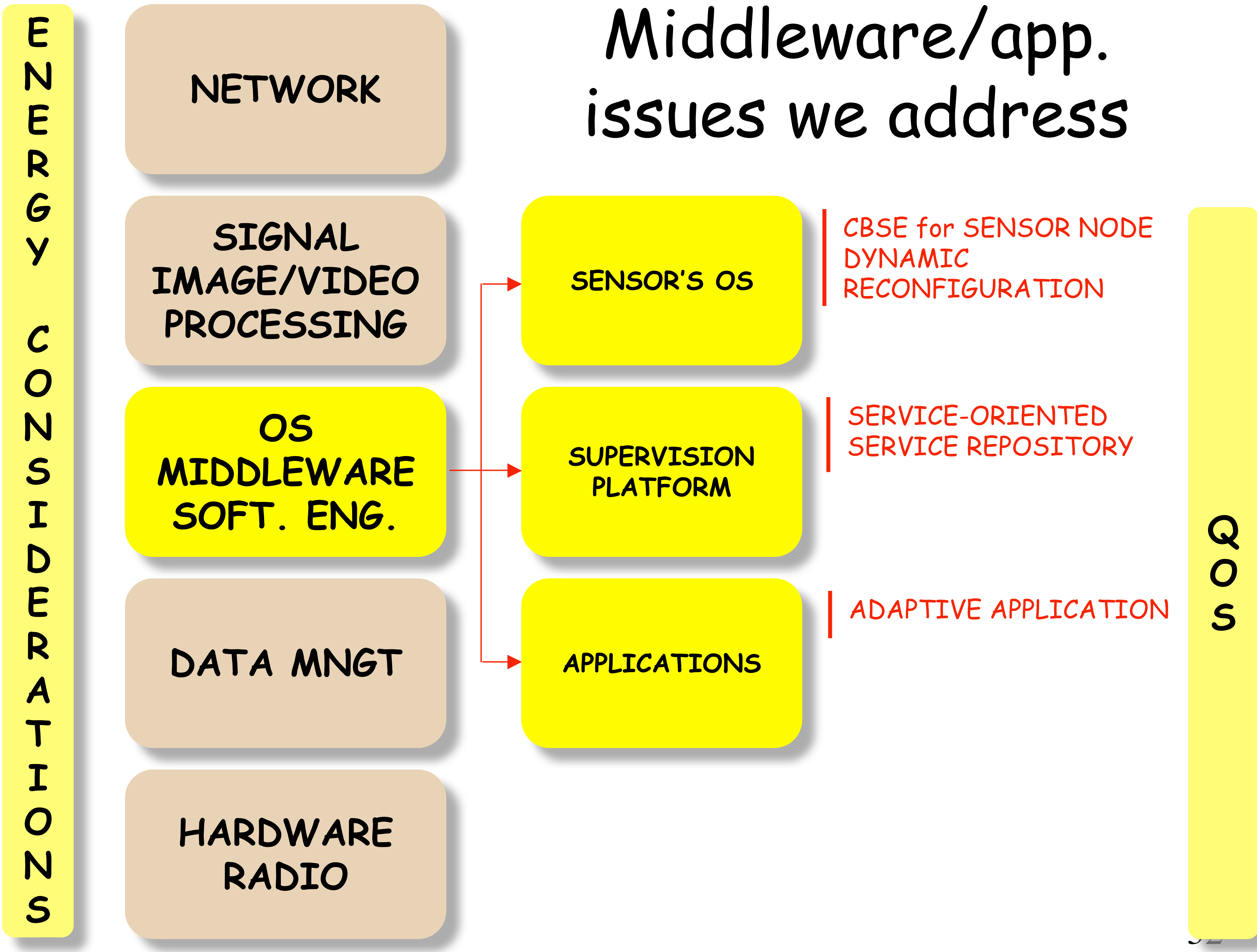


AVOIDS THE BLACK-BOX VISION

Research issues in WSN



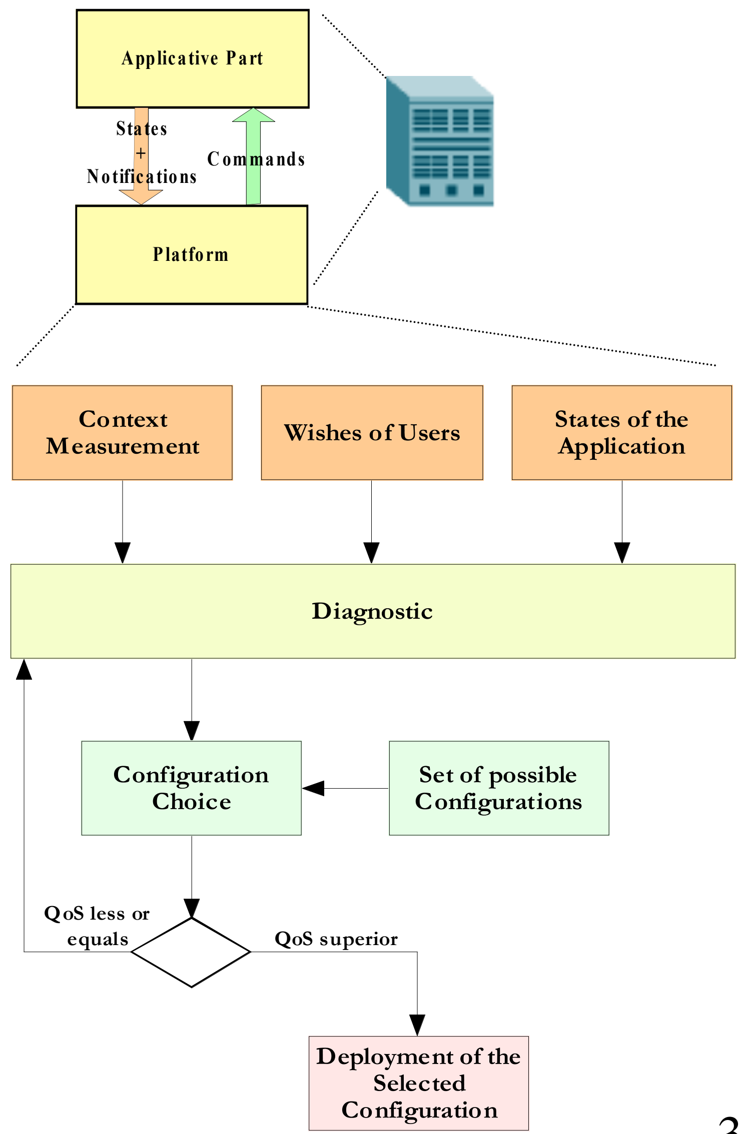
Middleware/app. issues we address



Supervision platform

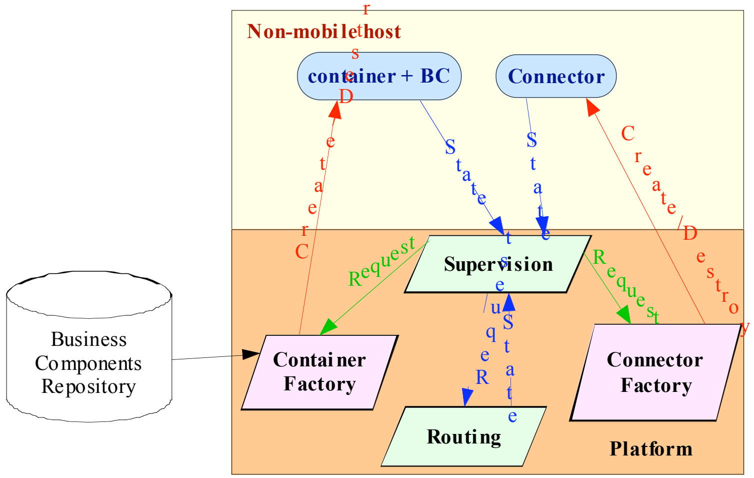
M. Dalmau & P. Roose

- Take care of user's QoS and QoS continuity
- Allows for a service-oriented surveillance system
- Discovery and publish mechanisms
- In charge of determining which configuration is better

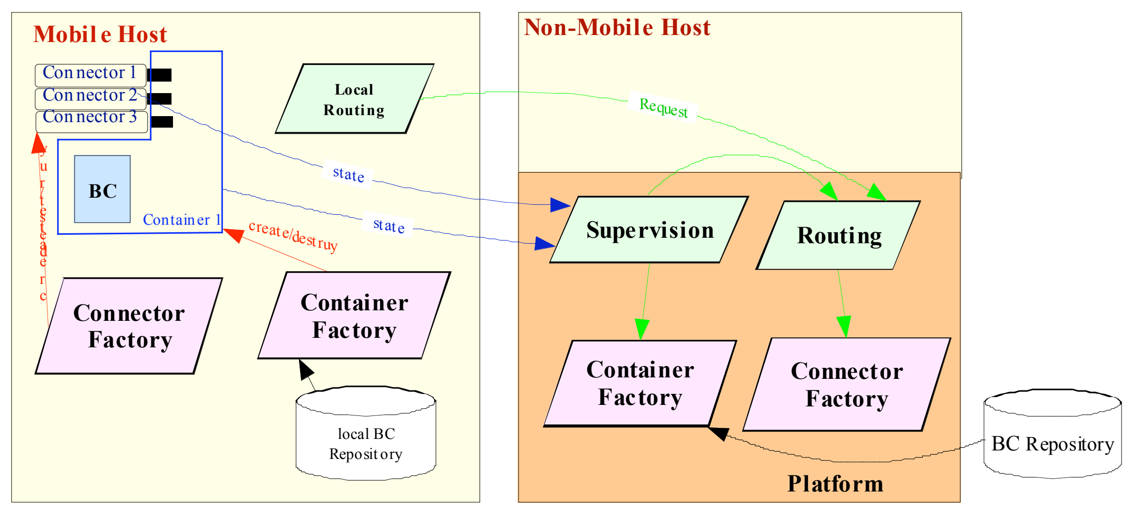


A bit of the internal design

Fixed-node/base station



Mobile/lightweight-node



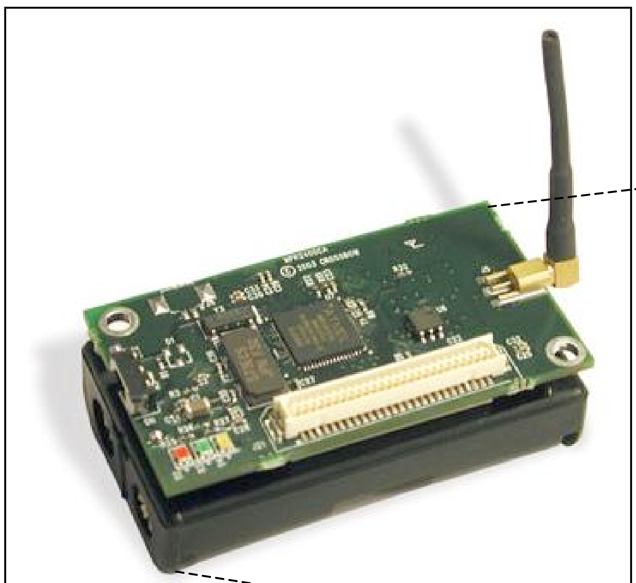
Dynamic reconfiguration (1)

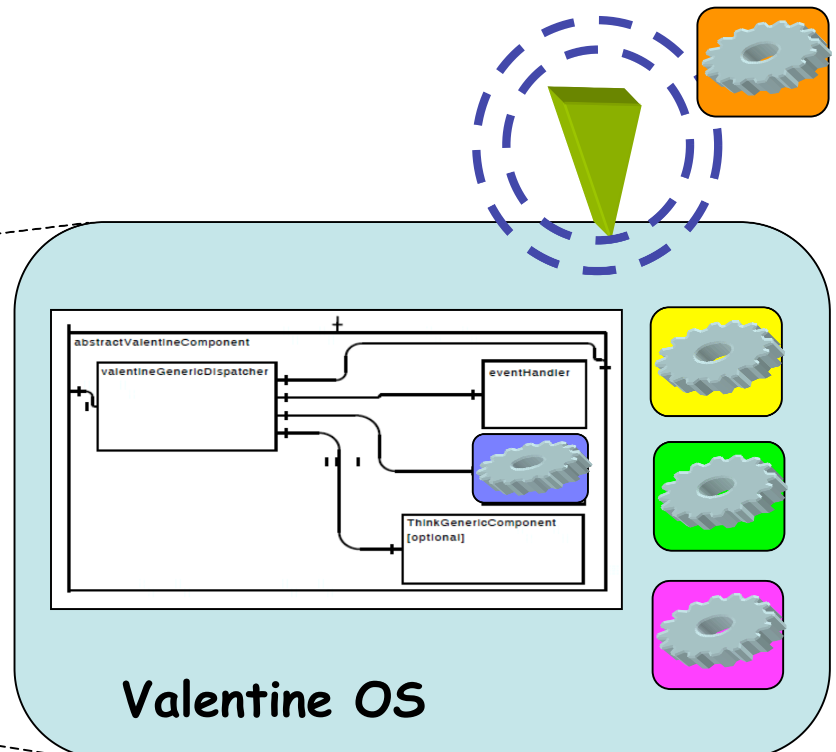
N. Hoang (PhD student)

- Avoids monolithic OS (à la TinyOS)
- We use the Think framework, which is an implementation of the Fractal component model to generate dynamic and reconfigurable OS and services
- First step towards the « active and programmable networking » concept applied to Wireless Sensor Networks

Dynamic reconfiguration (2)

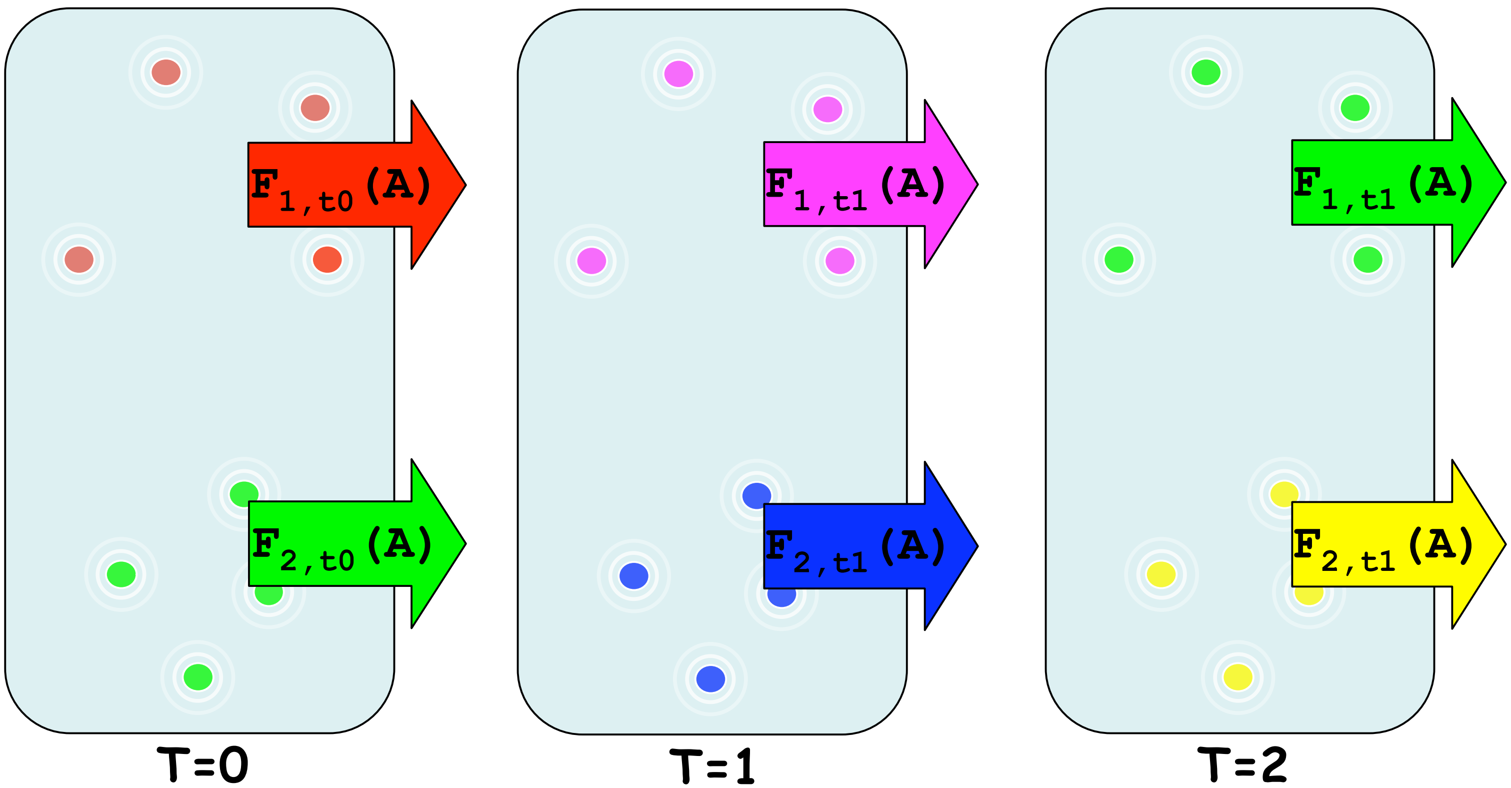
- Target platform: MicaZ
- Extension of the Think generic components
  ➔Valentine OS



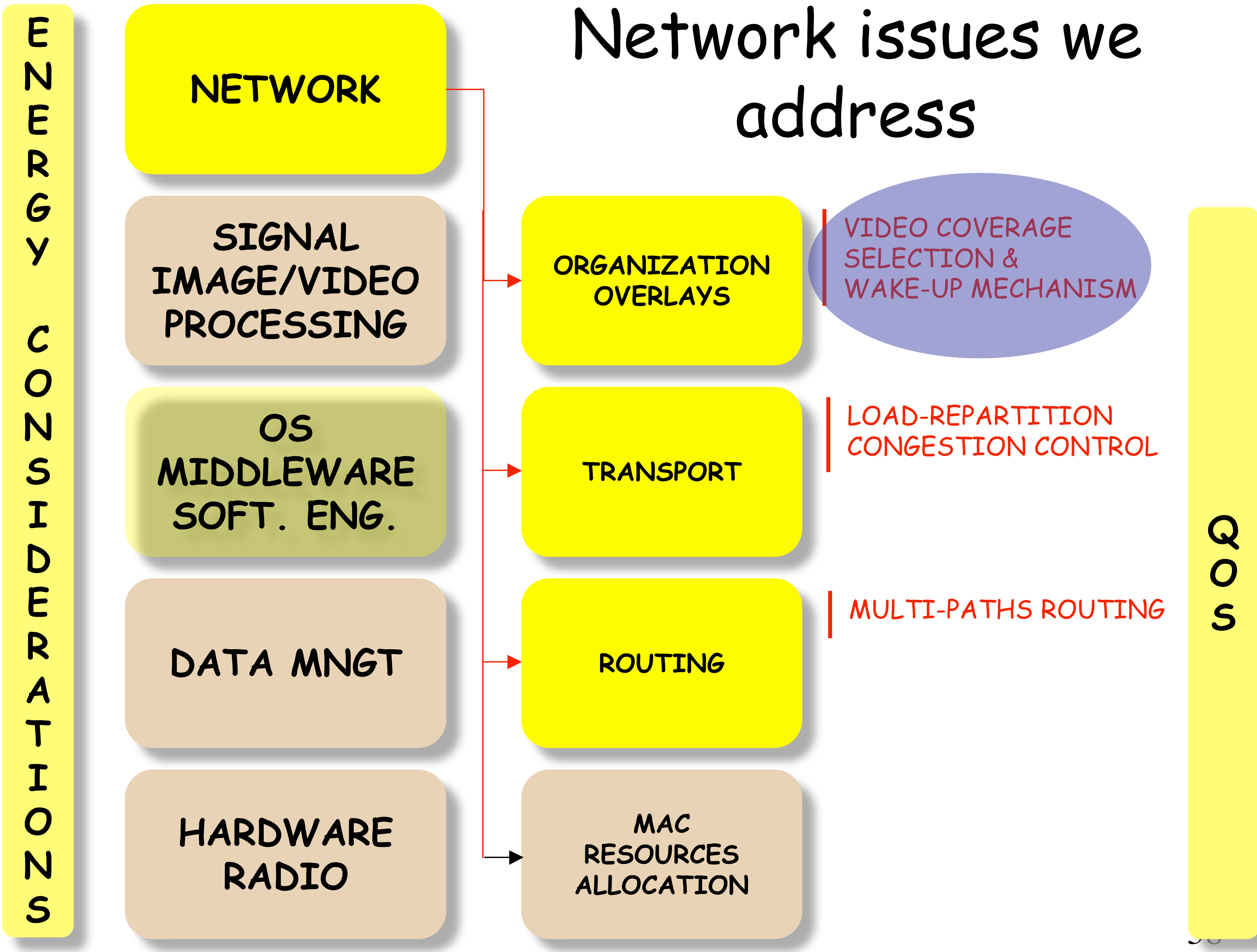


Towards Service Oriented Architecture

- Fast reconfiguration enables dynamic and on-the-fly new services deployment

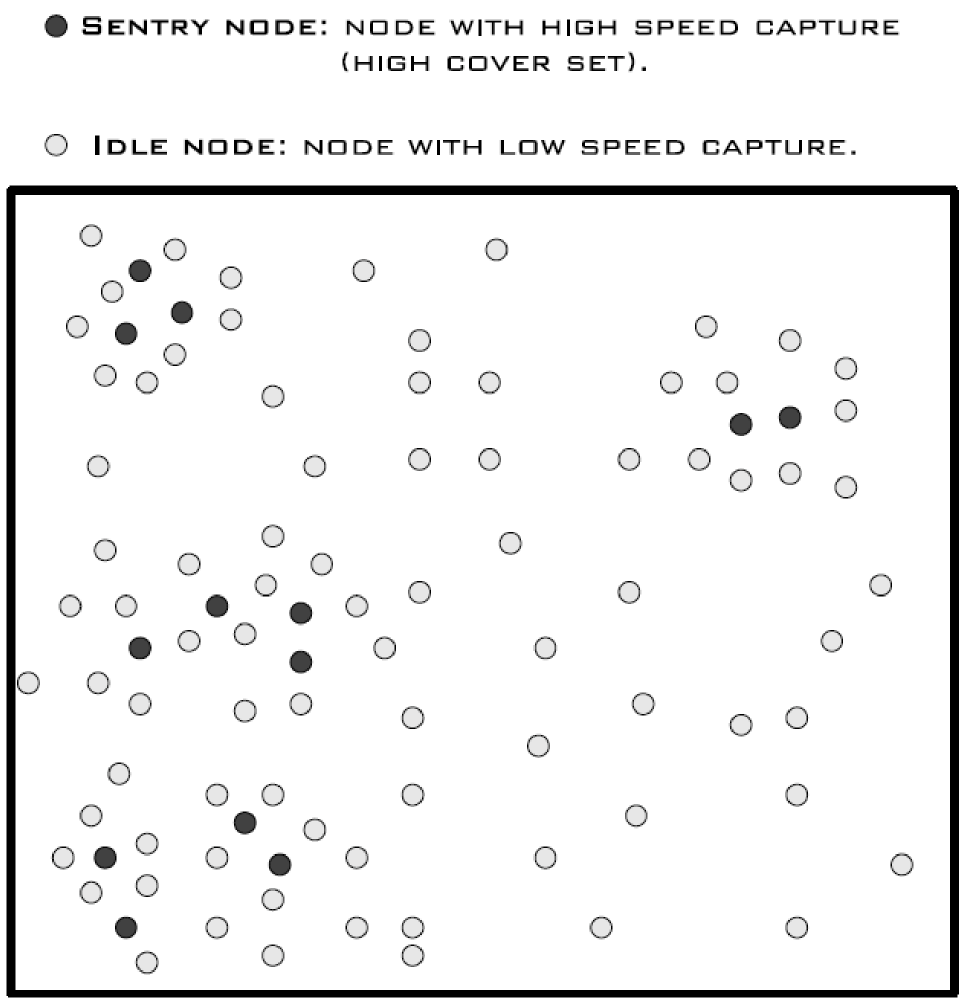


Network issues we address



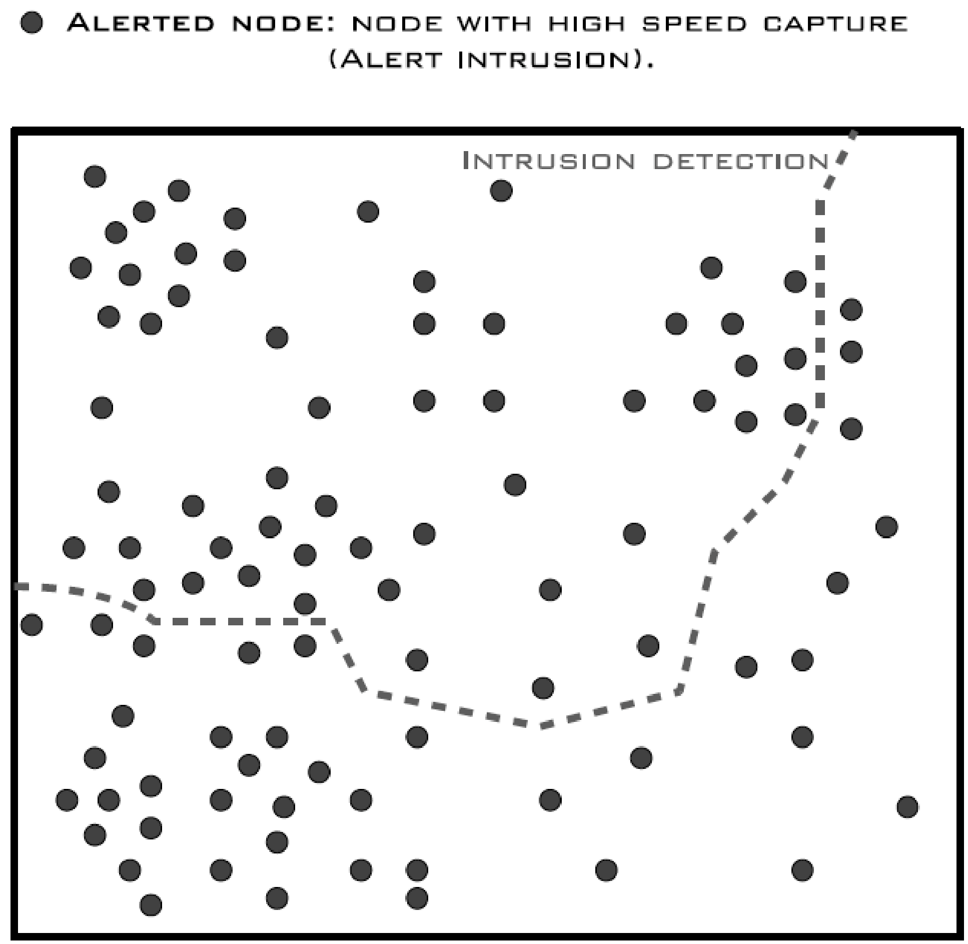
Surveillance scenario (1)

- Randomly deployed video sensors
- Not only barrier coverage but general intrusion detection
- Most of the time, network in so-called *hibernate mode*
- Most of active sensor nodes in *idle mode* with low capture speed
- Sentry nodes with higher capture speed to quickly detect intrusions



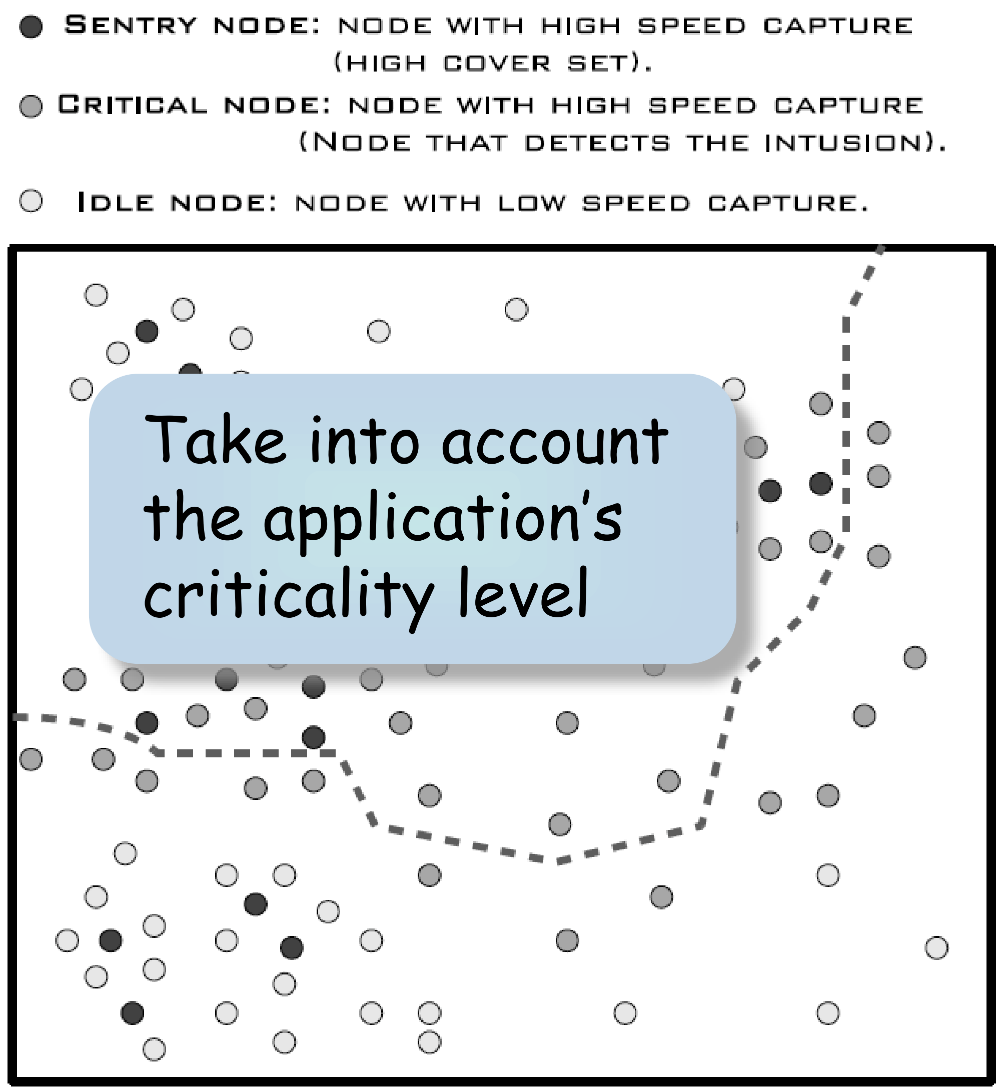
Surveillance scenario (2)

- Nodes detecting intrusion must alert the rest of the network
- 1-hop to k-hop alert
- Network in so-called *alerted mode*
- Capture speed must be increased
- Ressources should be focused on making tracking of intruders easier



Surveillance scenario (3)

- Network should go back to *hibernate mode*
- Nodes on the intrusion path must keep a high capture speed
- Sentry nodes with higher capture speed to quickly detect intrusions



Application's criticality

- All surveillance applications may not have the same criticality level, $r^0 \in [0,1]$
  - Environmental, security, healthcare,...
- Capture speed should decrease when r^0 decreases
- Sensor nodes could be initialized with a given r^0 prior to deployment

How to meet app's criticality

- Capture speed can be a « quality » parameter
- Capture speed for node v should depend on the app's criticality and on the level of redundancy for node v
- V's capture speed can increase when as V has more nodes covering its own FoV - cover set

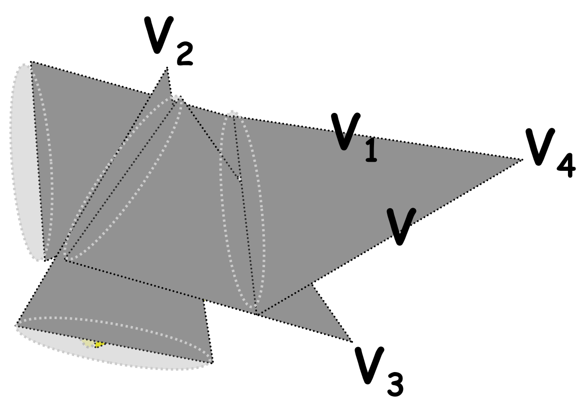
Video Sensor Nodes

Imote2

Multimedia board

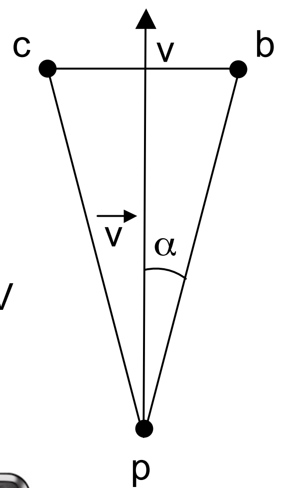
Node's cover set

- Each node v has a Field of View, FoV_v
- $Co_i(v)$ = set of nodes v' such as $\bigcup_{v' \in Co_i(v)} FoV_{v'}$ covers FoV_v
- $Co(v)$= set of $Co_i(v)$



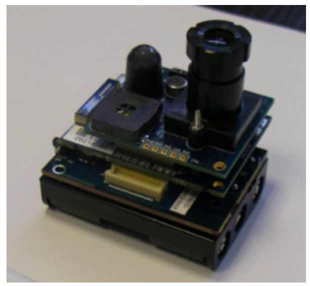
$Co(v)=\{V_1,V_2,V_3,V_4\}$

Finding v's cover set

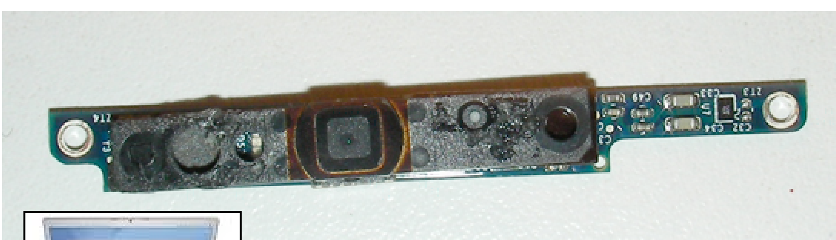


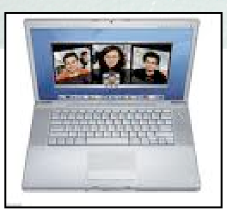
2α=AoV

AoV=20°



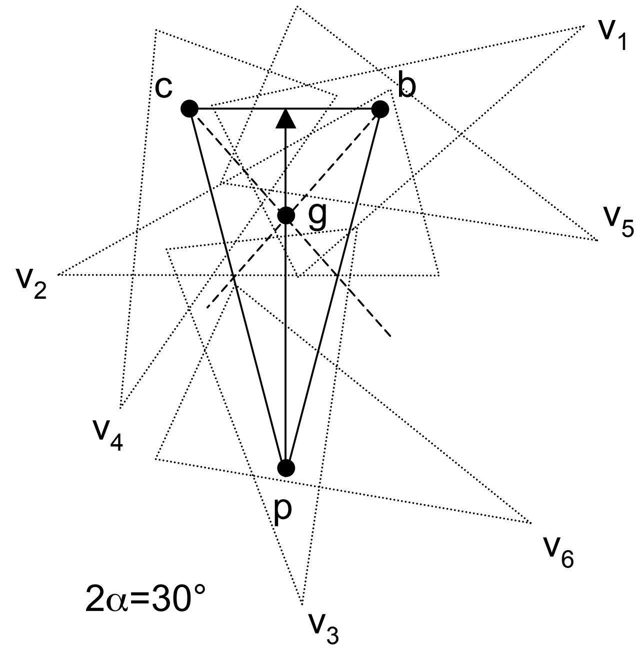
AoV=38°





AoV=31°

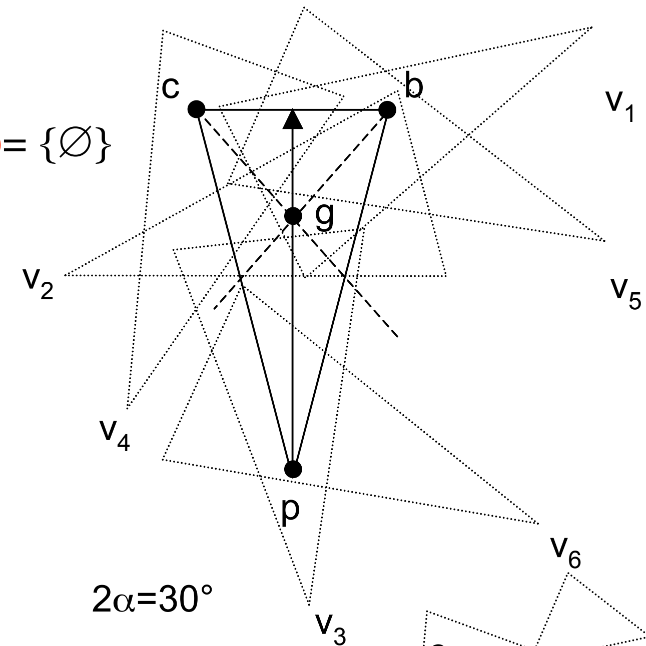
P = {v ∈ N(V) : v covers the point "p" of the FoV}
B = {v ∈ N(V) : v covers the point "b" of the FoV}
C = {v ∈ N(V) : v covers the point "c" of the FoV}
G = {v ∈ N(V) : v covers the point "g" of the FoV}

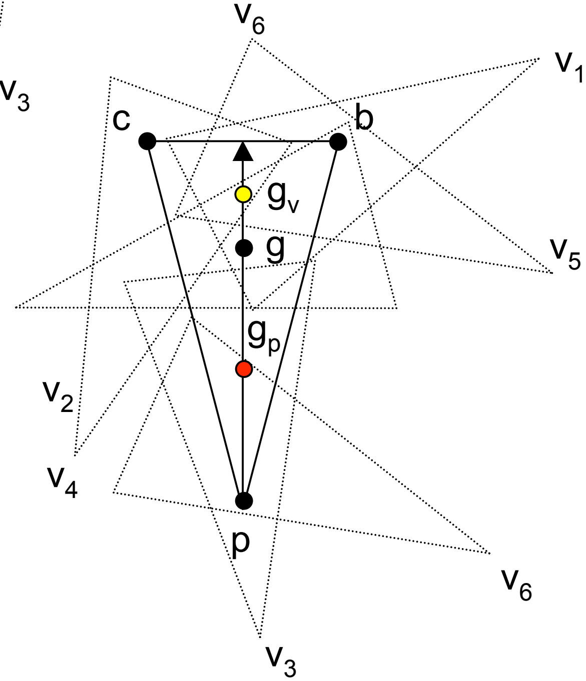


PG={P∩G}
BG={B∩G}
CG={C∩G}
Co(v)=AG×BG×CG

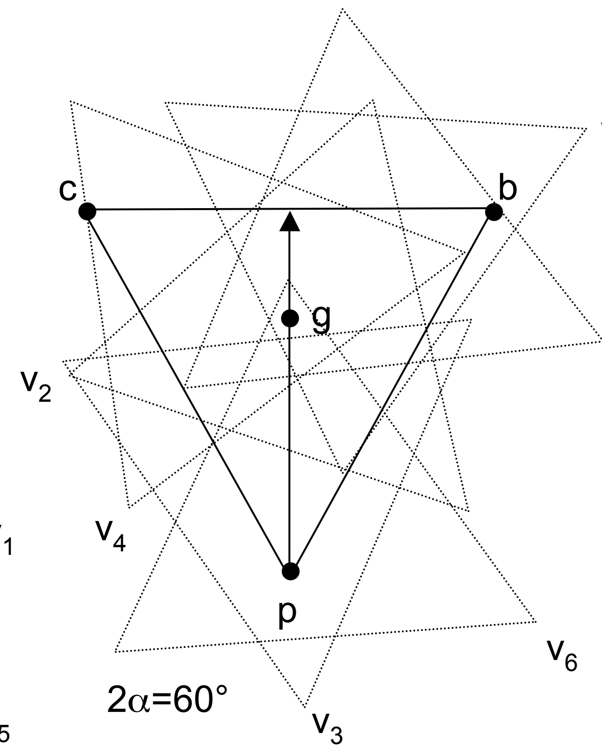
Small Angle of View

Co(V)= {∅}



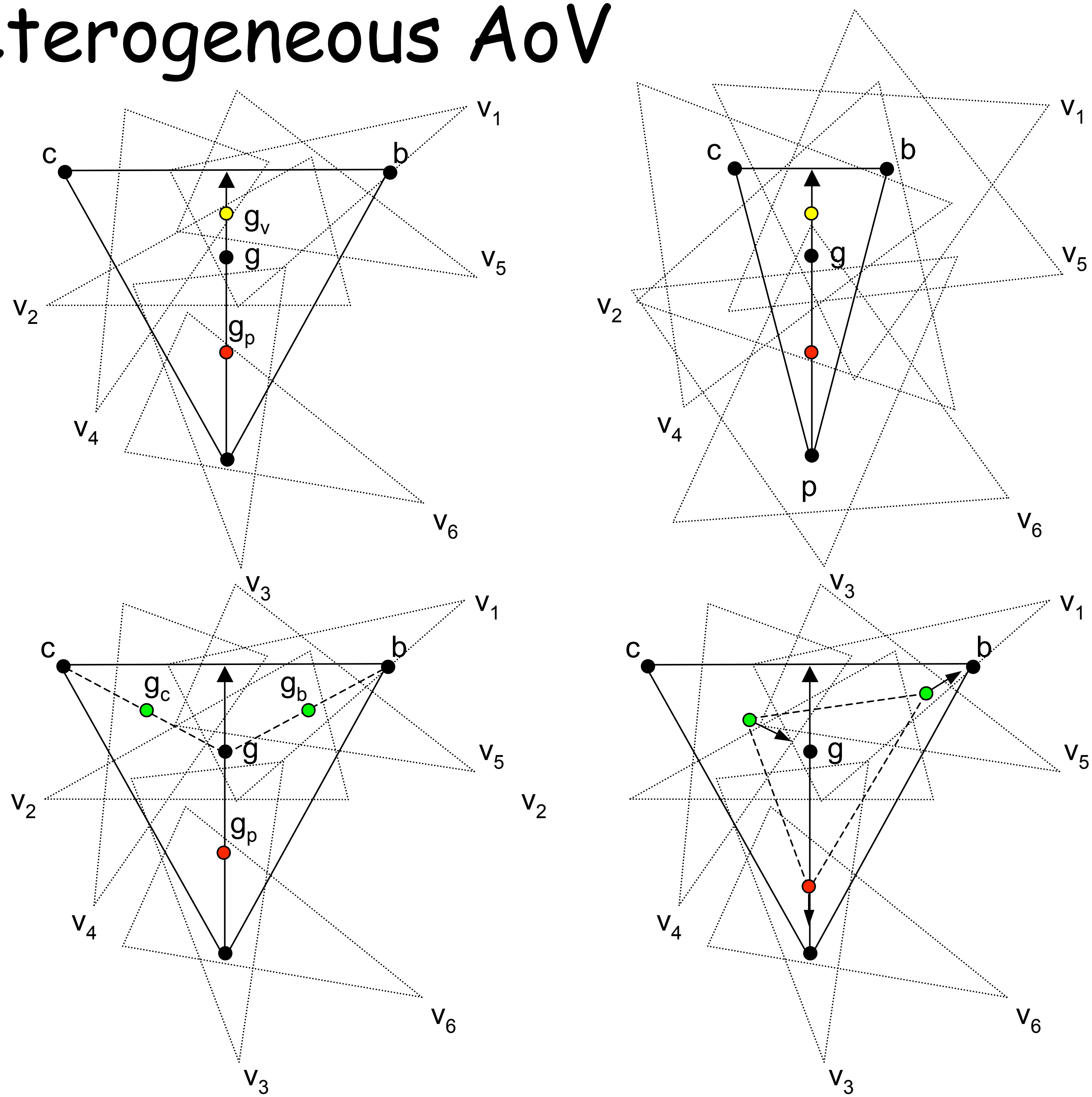


Co(V)= {
{V },
{V_1, V_3, V_4},
{V_2, V_3, V_4},
{V_3, V_4, V_5},
{V_1, V_4, V_6},
{V_2, V_4, V_6},
{V_4, V_5, V_6}
}

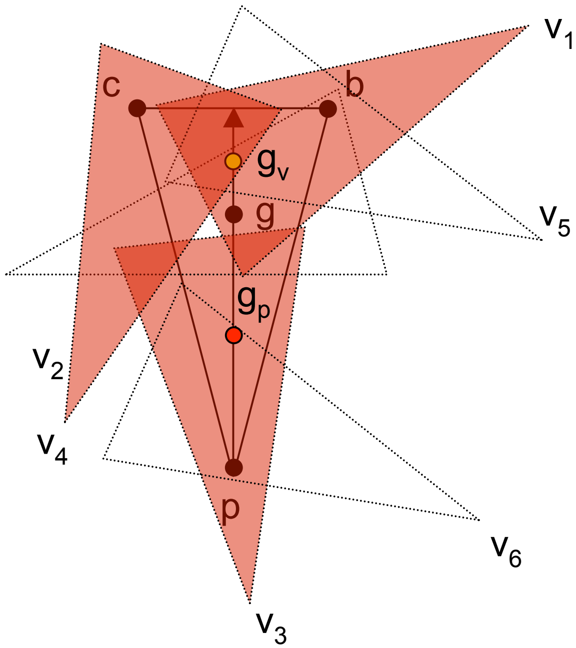


Co(V)= {
{V },
{V_1, V_4, V_6},
{V_4, V_5, V_6}
}

Heterogeneous AoV



Accuracy of cover set (1)



Co(V)= {
{V },
{V_1, V_3, V_4},
{V_2, V_3, V_4},
{V_3, V_4, V_5},
{V_1, V_4, V_6},
{V_2, V_4, V_6},
{V_4, V_5, V_6}
}

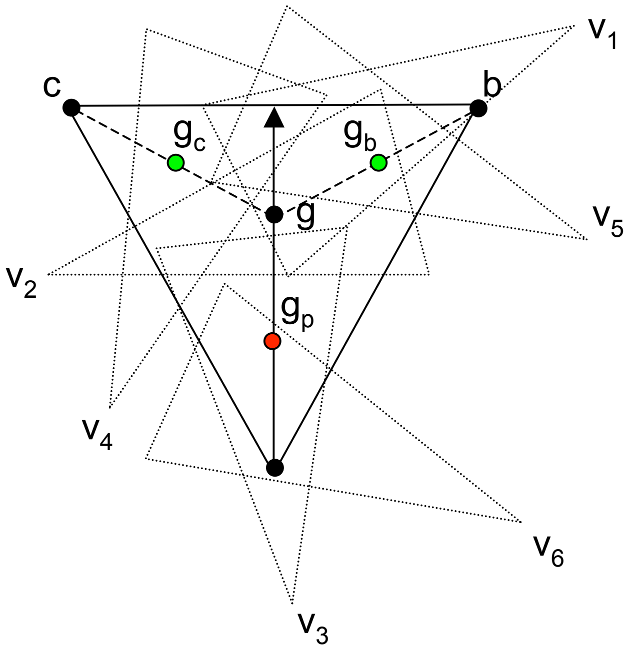
COV_{woG} 36° #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	38.22	46.96	14.40,86.27	18.08	1,45.33	9.59
100	49.67	47.67	16.58,83.05	15.53	1.33,110.66	20.31
125	64.80	48.67	15.48,89.40	15.67	1,150	29.98
150	67.78	48.81	12.77,90.88	15.67	1.66,170.33	35.93
175	75.24	47.83	18.50,89.56	13.31	1.66,412	82.24

COV_{wG} 36° #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	0	0	0,0	nan	0,0	0
100	0.67	65.64	65.64,65.64	0	1,1	1
125	1.87	91.45	88.83,93.15	2.97	1.33,2	1.56
150	1.78	95.06	91.47,98.29	4.06	1,3	1.94
175	3.43	94.42	87.60,99.03	4.40	1.33,2.66	1.92

COV_{waGpv} 36° #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	6.22	82.07	74.78,89.98	6.24	1.33,4	2.23
100	11	79.22	55.47,96.68	13.16	1,5.33	2.05
125	18.93	79.86	49.99,98.90	12.14	1,11.33	3.23
150	18.89	82.22	54.56,99.07	11.67	1,8.66	2.97
175	26.67	82.07	59.26,99.26	10.17	1,22.66	5.32

COV_{waGbc} 36° #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	12.44	77.48	56.46,91.81	13.33	1.33,9.33	3.62
100	20.33	79.62	53.65,98.98	12.05	1,10.66	3.94
125	30.67	76.89	50.53,97.92	11.58	1,34	5.40
150	35.11	78.47	52.07,96.09	10.60	1,31.33	6.90
175	48.57	77.76	49.97,98.20	10.54	1,50.33	11.57

Accuracy of cover set (2)



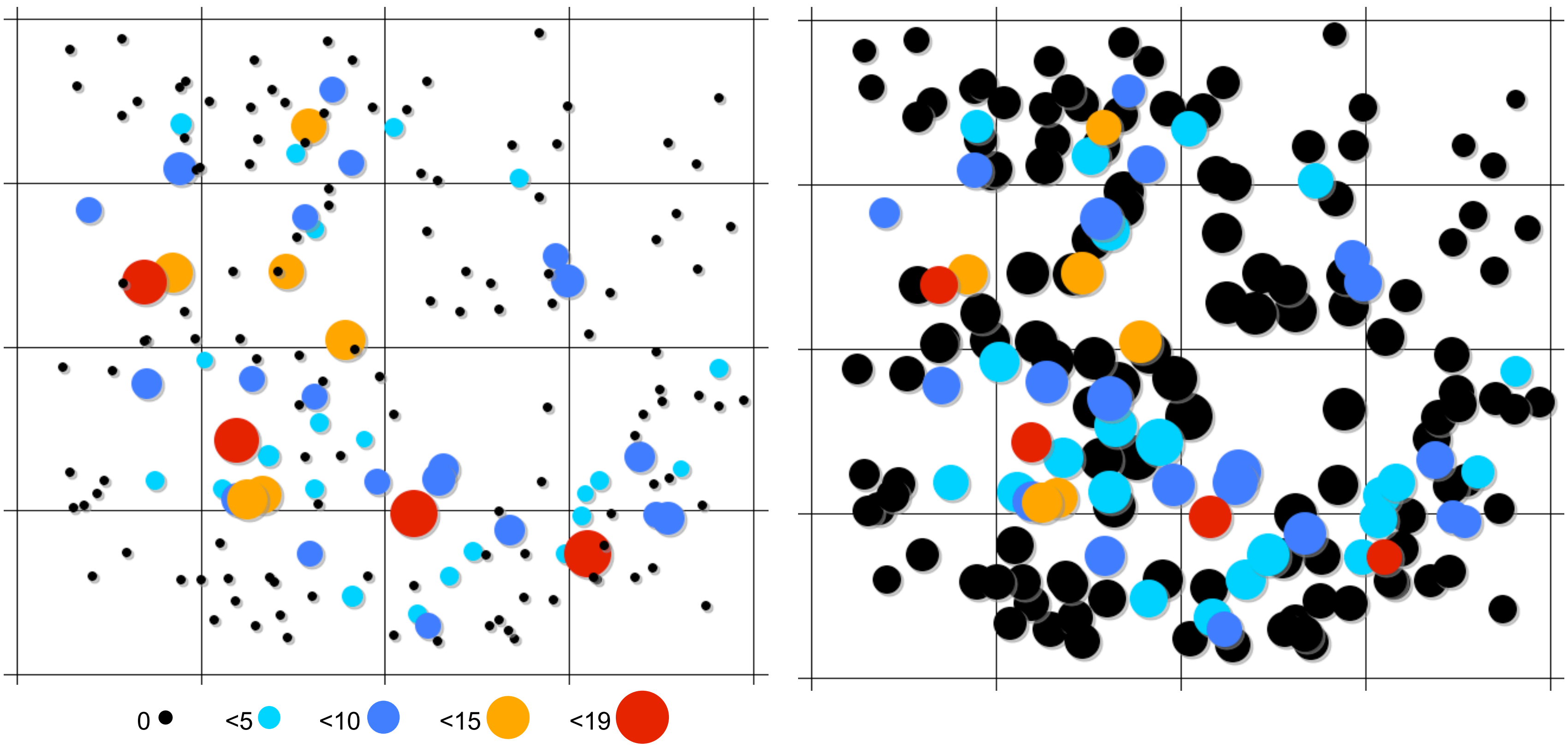
COV_{waGpv} $36^o(50\%)$ $60^o(50\%)$ #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	11.56	83.36	70.20,93.99	9.12	1,8	2.70
100	16.33	86.88	61.52,99.50	11.21	1,13.33	3.62
125	29.07	89.07	63.14,100	9.20	1,24.66	6.66
150	33.56	88.01	56.18,99.99	10.06	1,40	8.23
175	43.81	88.52	58.76,99.97	9.02	1,45.33	10.47

COV_{waGbc} $36^o(50\%)$ $60^o(50\%)$ #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	8.44	85.81	71.60,96.59	10.22	1,5.66	2.53
100	12.33	79.34	56.33,94.49	12.08	1.33,14	4.92
125	13.87	80.88	61.50,94.87	9.63	1.33,35.33	10.27
150	18.22	76.04	54.17,97.23	11.81	1,34	9.58
175	24.95	75.21	55,92.26	9.33	1.66,99.33	18.93

COV_{waGpv} $36^o(80\%)$ $60^o(20\%)$ #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	16	81.97	60.34,100	11.84	1,9	2.83
100	15	88.34	69.60,100	9.00	1,12	3.13
125	14.40	85.16	55.43,100	14.14	1,12	4.17
150	28.67	85.95	57.58,100	10.88	1,16	3.77
175	33.14	85.94	54.34,100	11.85	1,32	6.21

COV_{waGbc} $36^o(80\%)$ $60^o(20\%)$ #nodes	% nodes with coverset	mean % coverage	min,max % coverage per coverset	stddev of % coverage	min,max #coverset per node	mean #coverset per node
75	10.67	83.39	57.20,97.34	14.34	2,12	5.38
100	17	86.29	62.58,99.78	12.44	1,12	3.06
125	41.60	81.41	56.86,95.58	8.36	1,48	11.25
150	47.33	81.92	55.51,100	11.18	1,48	9.39
175	54.86	80.18	51.84,98.24	10.41	1,120	17.20

Node's # of cover set

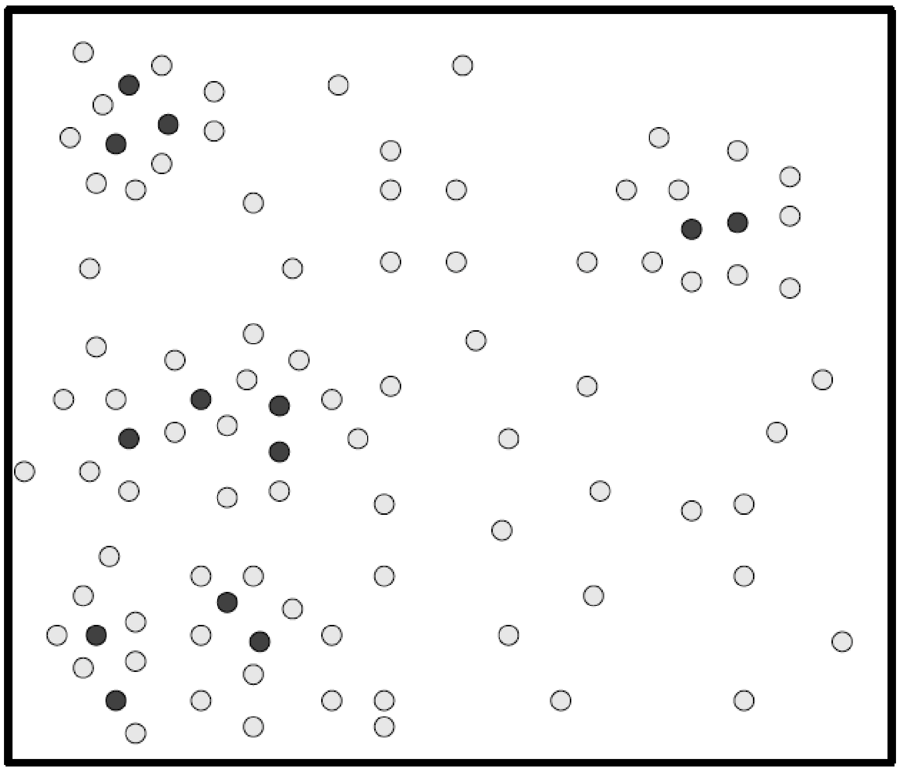


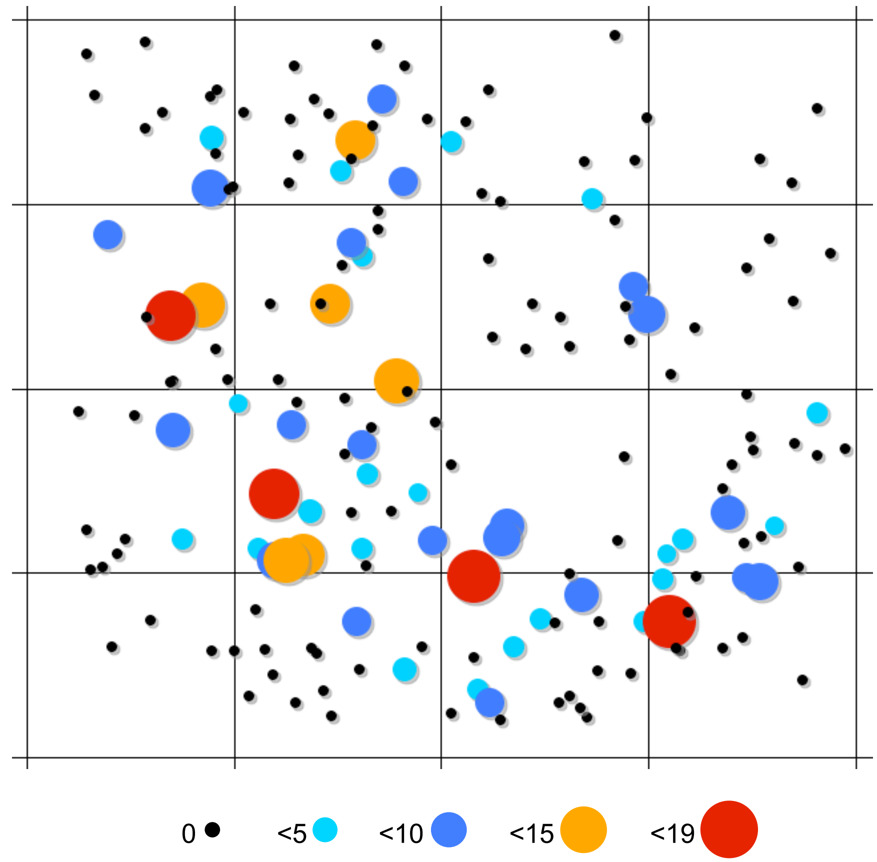
175 nodes, AoV=36°, COVwaGpv strategy

Defining sentry nodes

● Sentry node: node with high speed capture (high cover set).

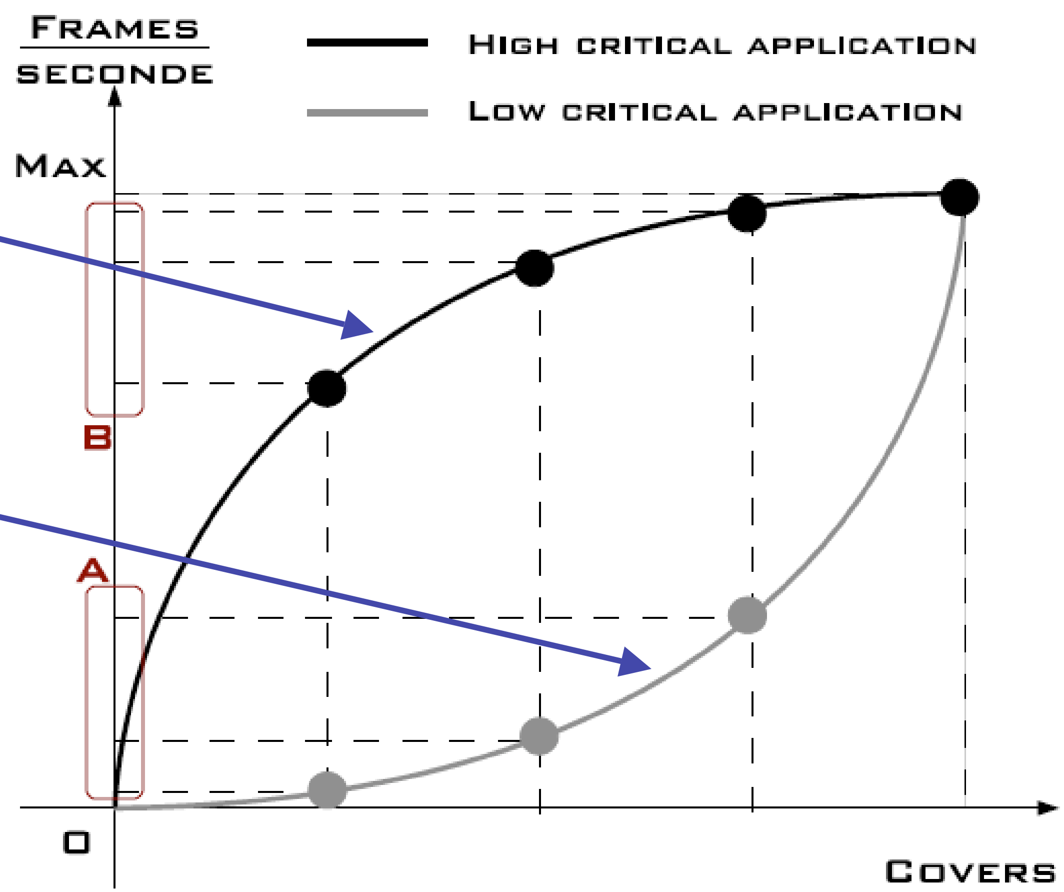
○ Idle node: node with low speed capture.





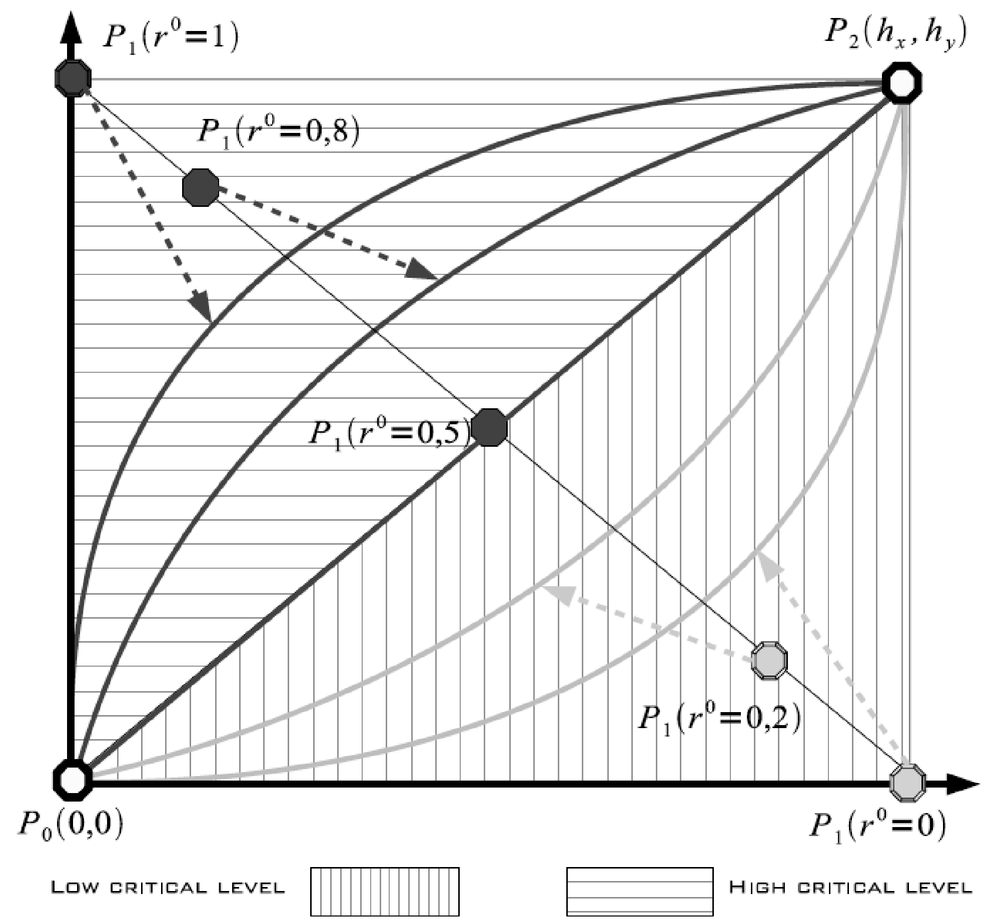
Criticality model (1)

- Link the capture rate to the size of the cover set
- High criticality
  - Convex shape
  - Most projections of x are close to the max capture speed
- Low criticality
  - Concave shape
  - Most projections of x are close to the min capture speed
- Concave and convex shapes automatically define sentry nodes in the network



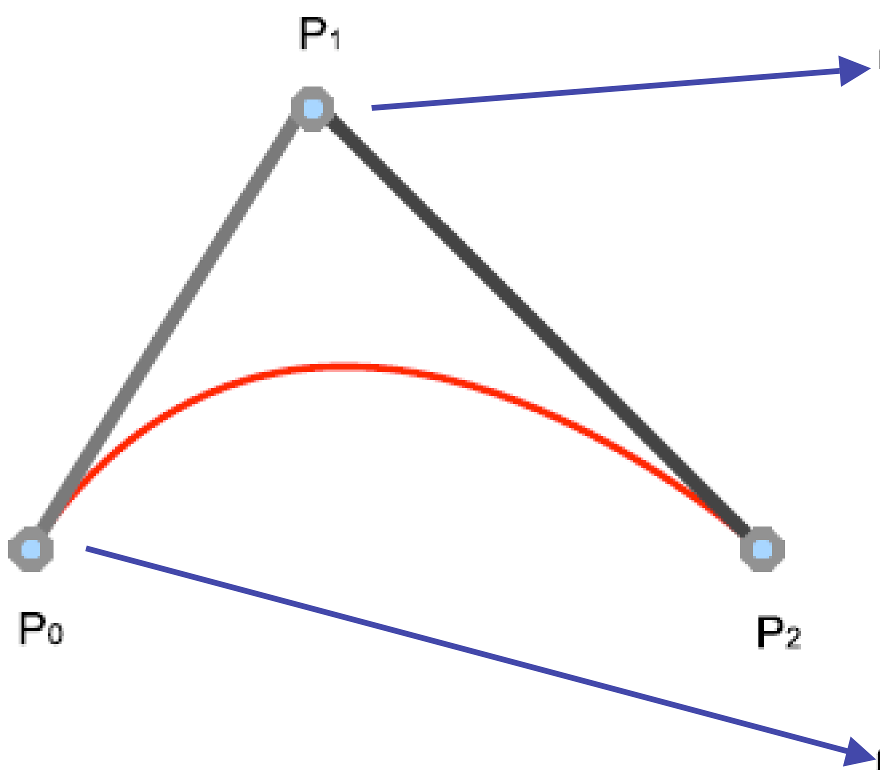
Criticality model (2)

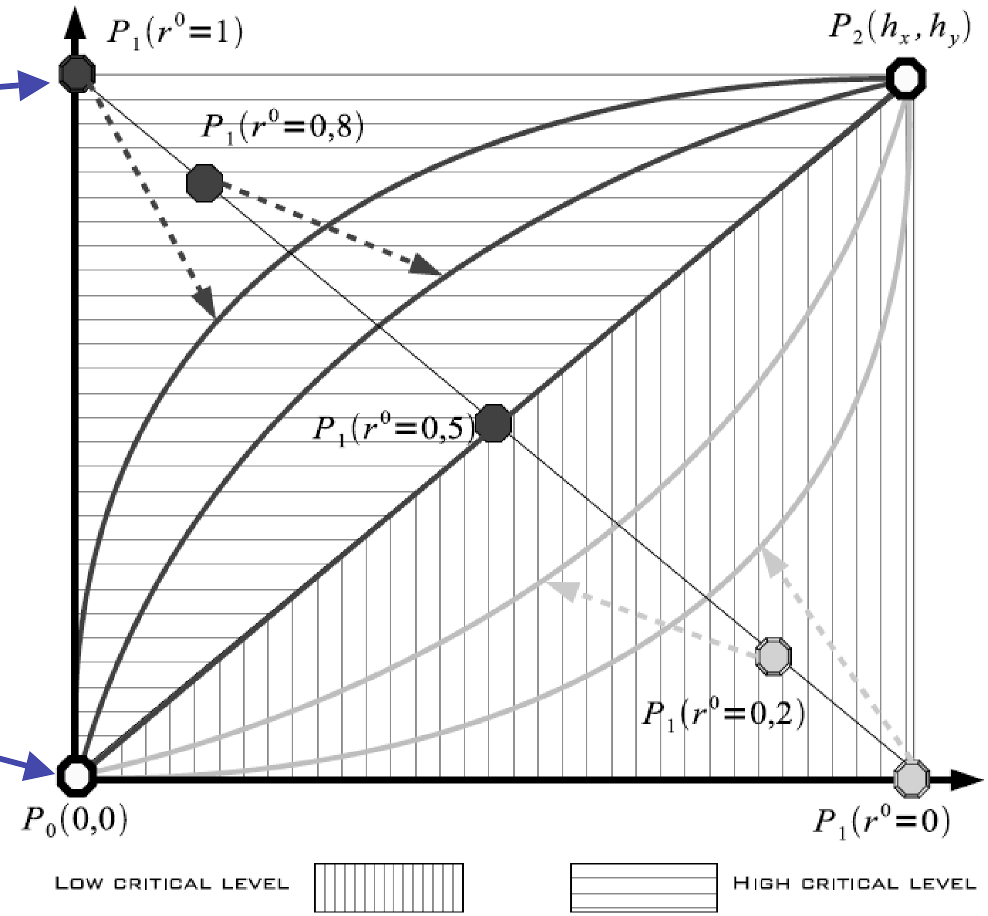
- r^0 can vary in [0,1]
- BehaVior functions (BV) defines the capture speed according to r^0
- $r^0 < 0.5$
  - Concave shape BV
- $r^0 > 0.5$
  - Convex shape BV
- We propose to use Bezier curves to model BV functions



BehaVior function

$$B(t) = (1-t)^2 * P_0 + 2t(1-t) * P_1 + t^2 * P_2$$





Some typical capture speed

- Maximum capture speed is 6fps
- Nodes with size of cover set greater than N capture at the maximum speed

N=6
$P_2(6,6)$

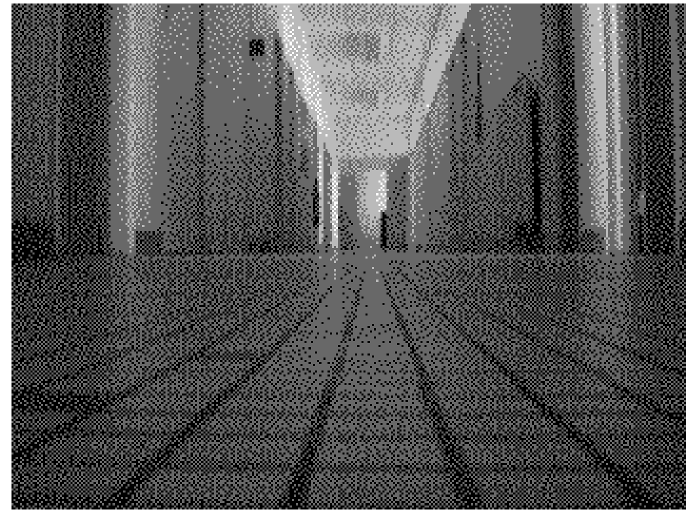
r^0 \ $\|Co(v)\|$	1	2	3	4	5	6
0.0	0.05	0.20	0.51	1.07	2.10	6.00
0.2	0.30	0.73	1.34	2.20	3.52	6.00
0.5	1.00	2.00	3.00	4.00	5.00	6.00
0.8	2.48	3.80	4.66	5.27	5.70	6.00
1.0	3.90	4.93	5.49	5.80	5.95	6.00

N=12
$P_2(12,3)$

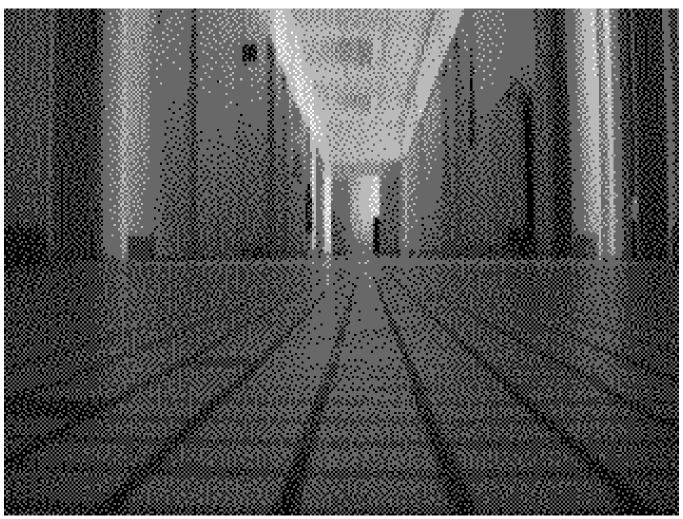
r^0	1	2	3	4	5	6	7	8	9	10	11	12
0	.01	.02	.05	0.1	.17	.26	.38	.54	.75	1.1	1.5	3
.2	.07	.15	.25	.37	.51	.67	.86	1.1	1.4	1.7	2.2	3
.4	.17	.35	.55	.75	.97	1.2	1.4	1.7	2.0	2.3	2.6	3
.6	.36	.69	1.0	1.3	1.5	1.8	2.0	2.2	2.4	2.6	2.8	3
.8	.75	1.2	1.6	1.9	2.1	2.3	2.5	2.6	2.7	2.8	2.9	3
1	1.5	1.9	2.2	2.4	2.6	2.7	2.8	2.9	2.9	2.9	2	3

320x200, 30 fps, 256 gray scale
15Mbps raw

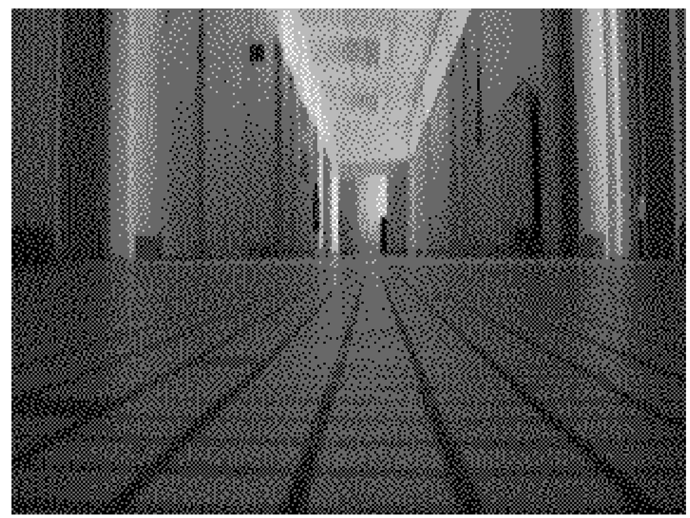
5 fps, 4 gray scale, 640kbps raw



2 fps, 4 gray scale, 256kbps raw



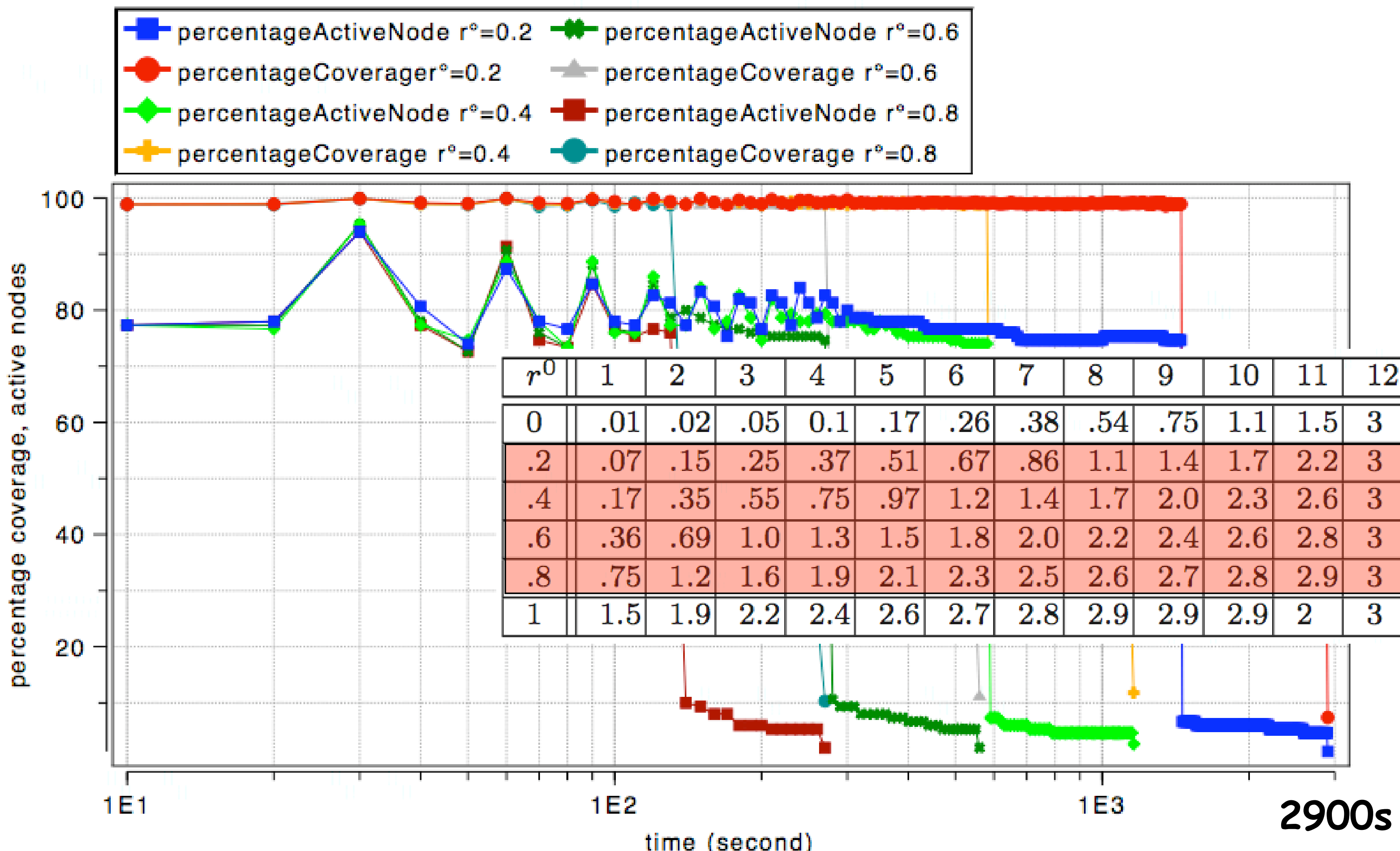
1 fps, 4 gray scale, 128kbps raw



Simulation settings

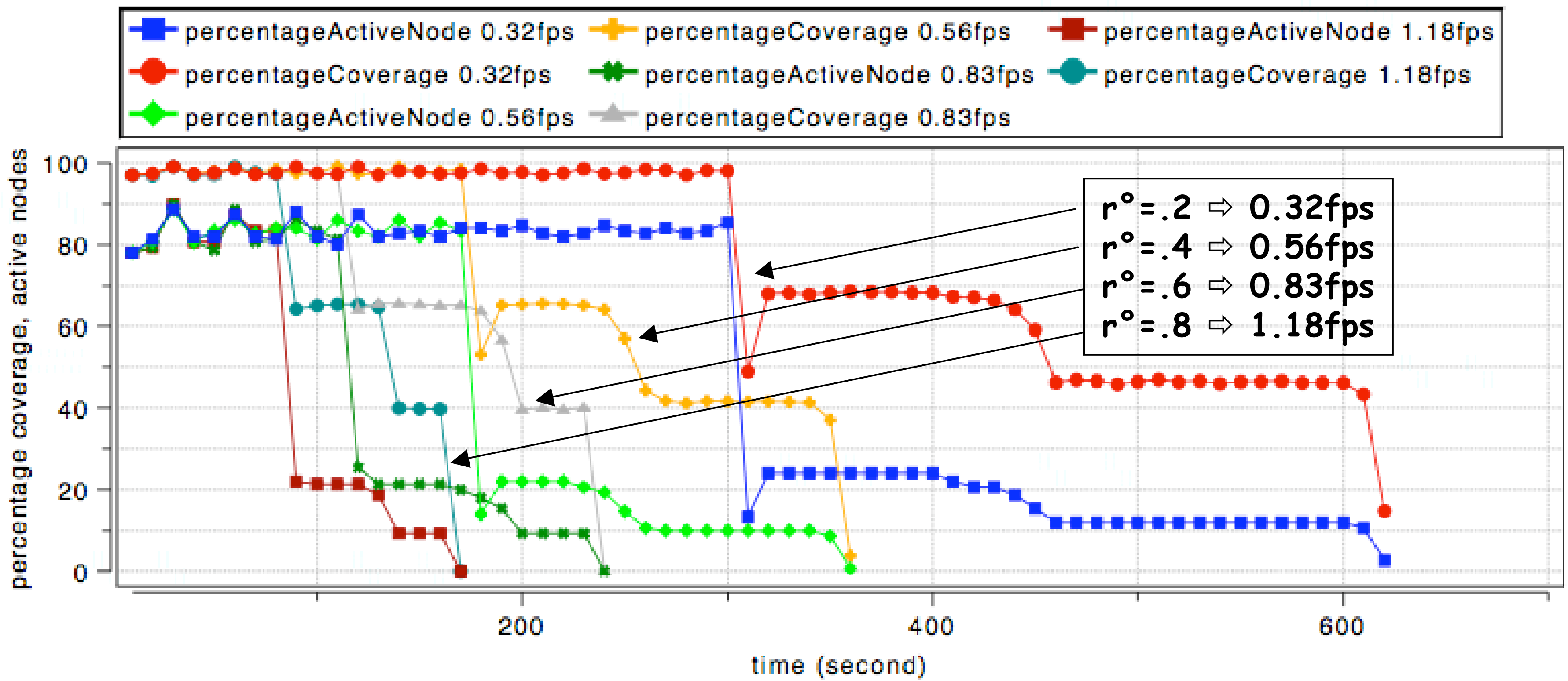
- OMNET++ simulation model
- Video nodes have communication range of 30m and depth of view of 25m, AoV is 36°. 150 sensors in an 75m.75m area.
- Battery has 100 units, 1 image = 1 unit of battery consumed.
- Max capture rate is 3fps. 12 levels of cover set.
- Full coverage is defined as the region initially covered when all nodes are active

Percentage of coverage, active nodes (1)



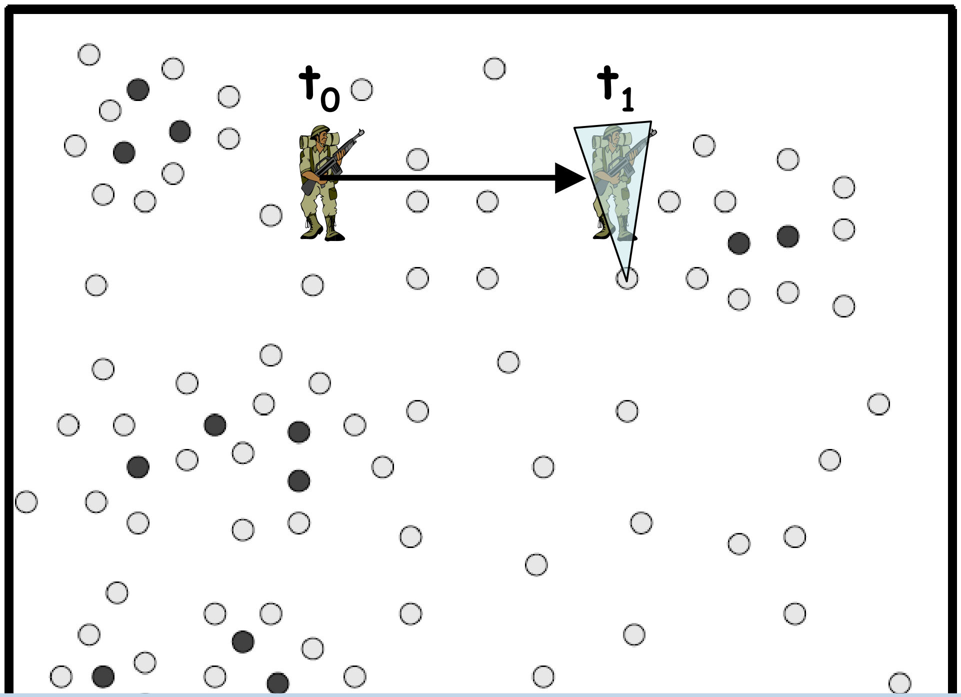
r^0	1	2	3	4	5	6	7	8	9	10	11	12
0	.01	.02	.05	0.1	.17	.26	.38	.54	.75	1.1	1.5	3
.2	.07	.15	.25	.37	.51	.67	.86	1.1	1.4	1.7	2.2	3
.4	.17	.35	.55	.75	.97	1.2	1.4	1.7	2.0	2.3	2.6	3
.6	.36	.69	1.0	1.3	1.5	1.8	2.0	2.2	2.4	2.6	2.8	3
.8	.75	1.2	1.6	1.9	2.1	2.3	2.5	2.6	2.7	2.8	2.9	3
1	1.5	1.9	2.2	2.4	2.6	2.7	2.8	2.9	2.9	2.9	2	3

Percentage of coverage, active nodes (2)



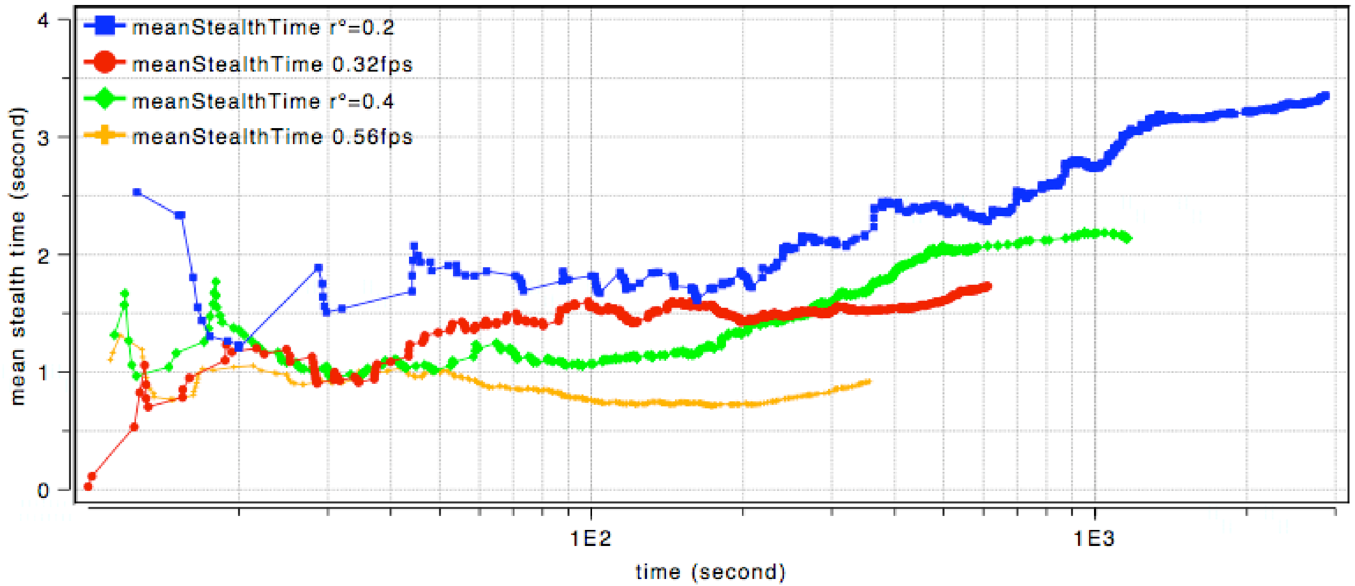
mean stealth time

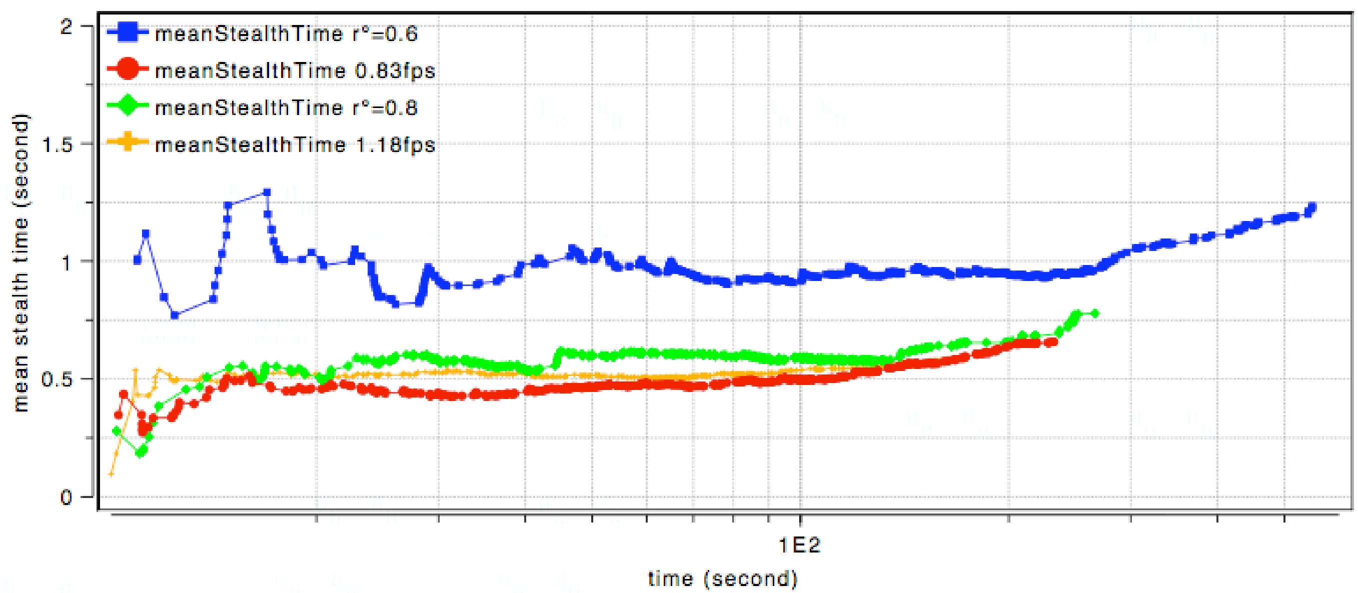
t_1-t_0 is the intruder's stealth time
velocity is set to 5m/s



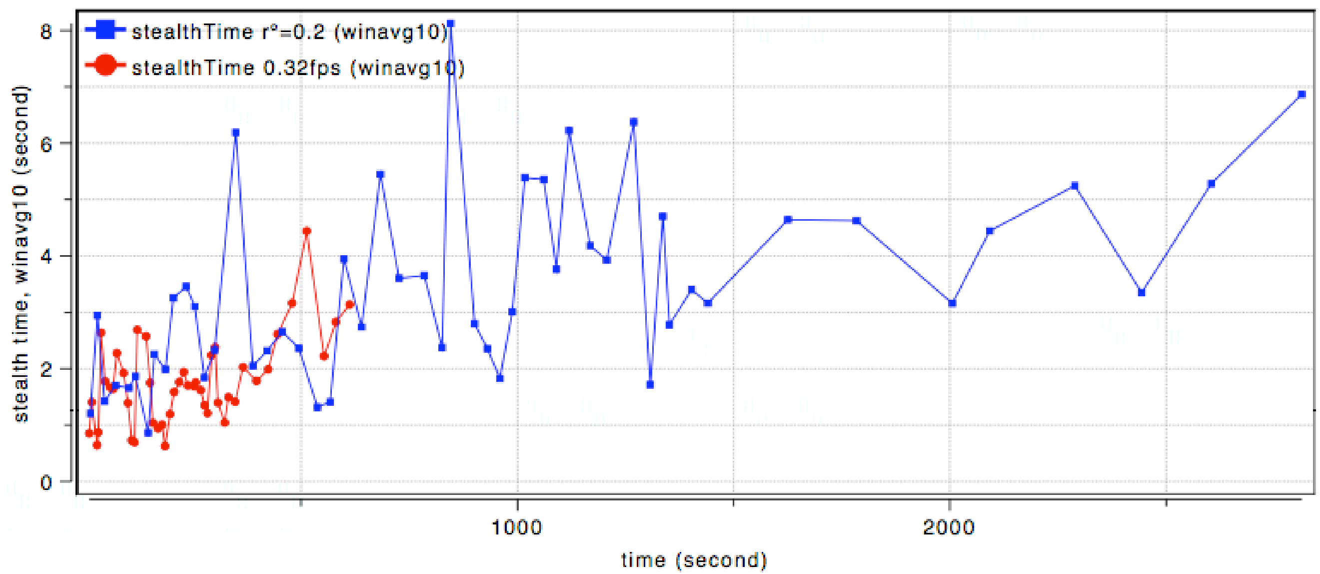
intrusions starts at t=10s
when an intruder is seen, compute the stealth time, and starts a new intrusion until end of simulation

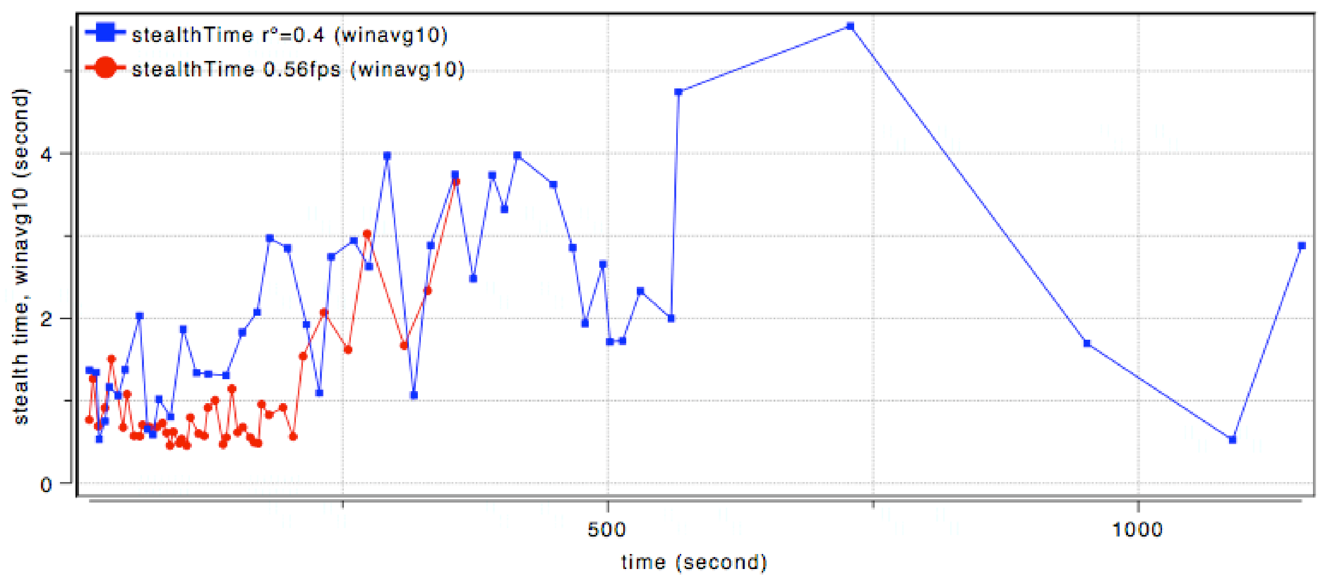
mean stealth time



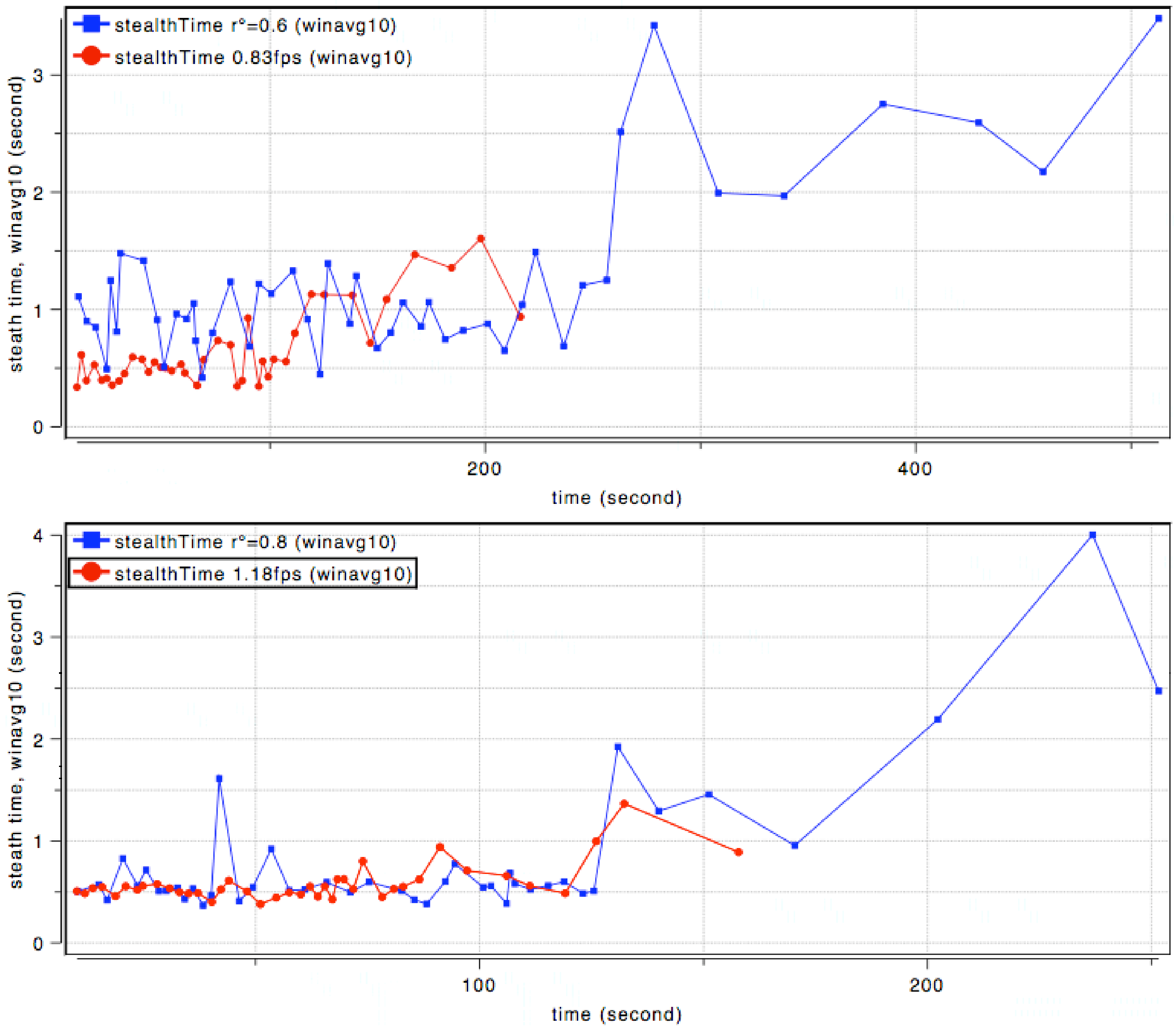


stealth time, winavg[10]



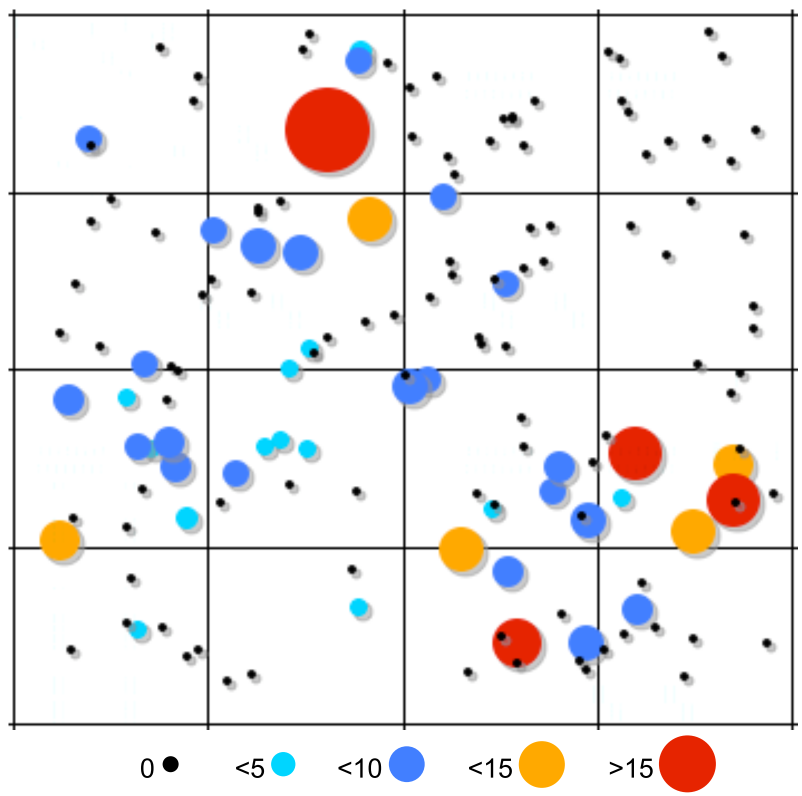


stealth time, winavg[10]

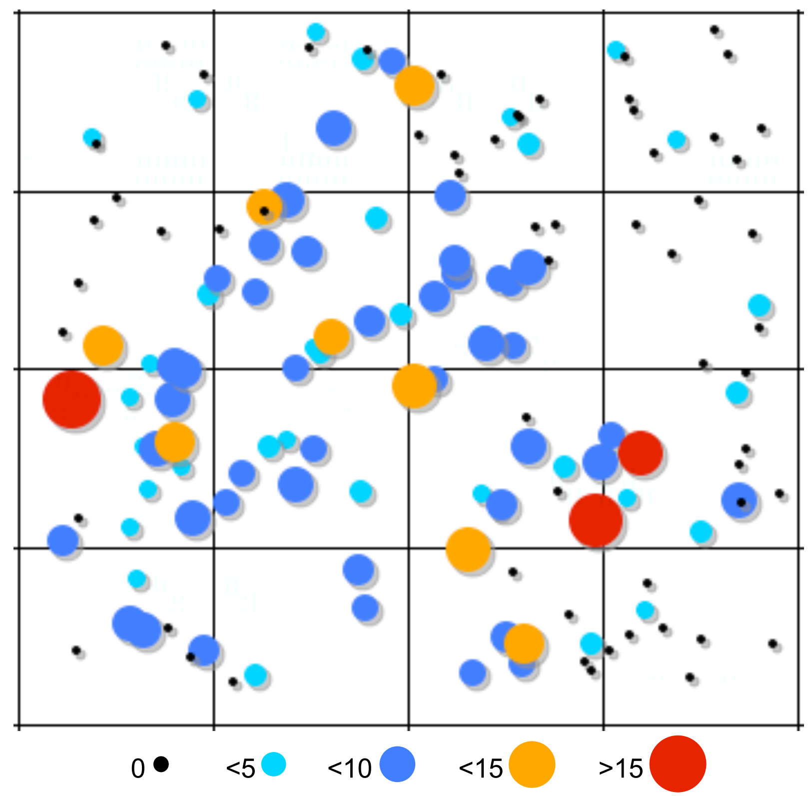


Sentry nodes

# of cover sets



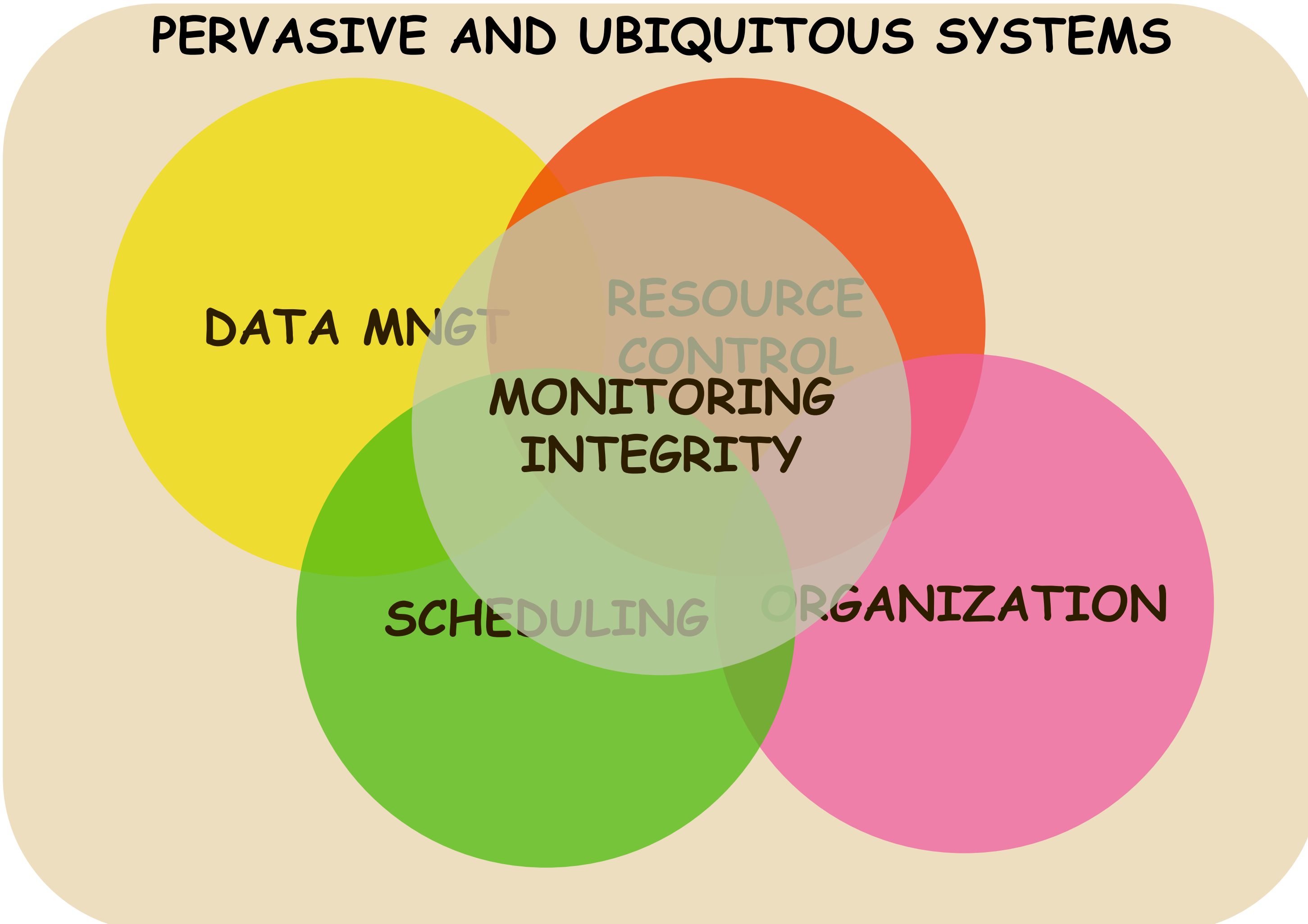
 # intrusion detected



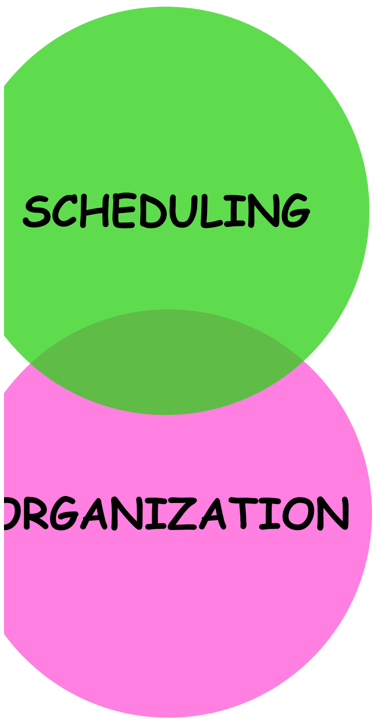
Conclusions

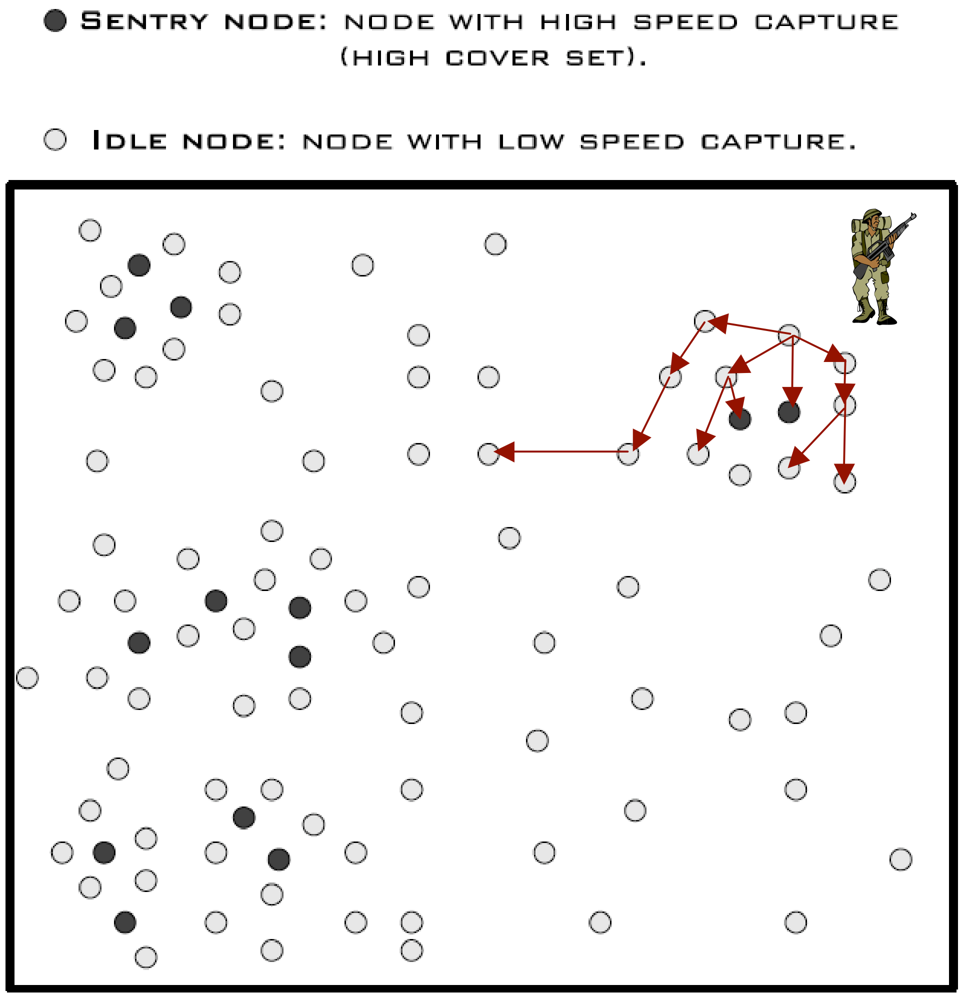
- Surveillance applications have a high level of criticity which make accountability important
- Criticality model with adaptive scheduling of nodes
- Optimize the resource usage by dynamically adjusting the provided service level
- Extension for risk-based scheduling in intrusion detection systems

Research directions

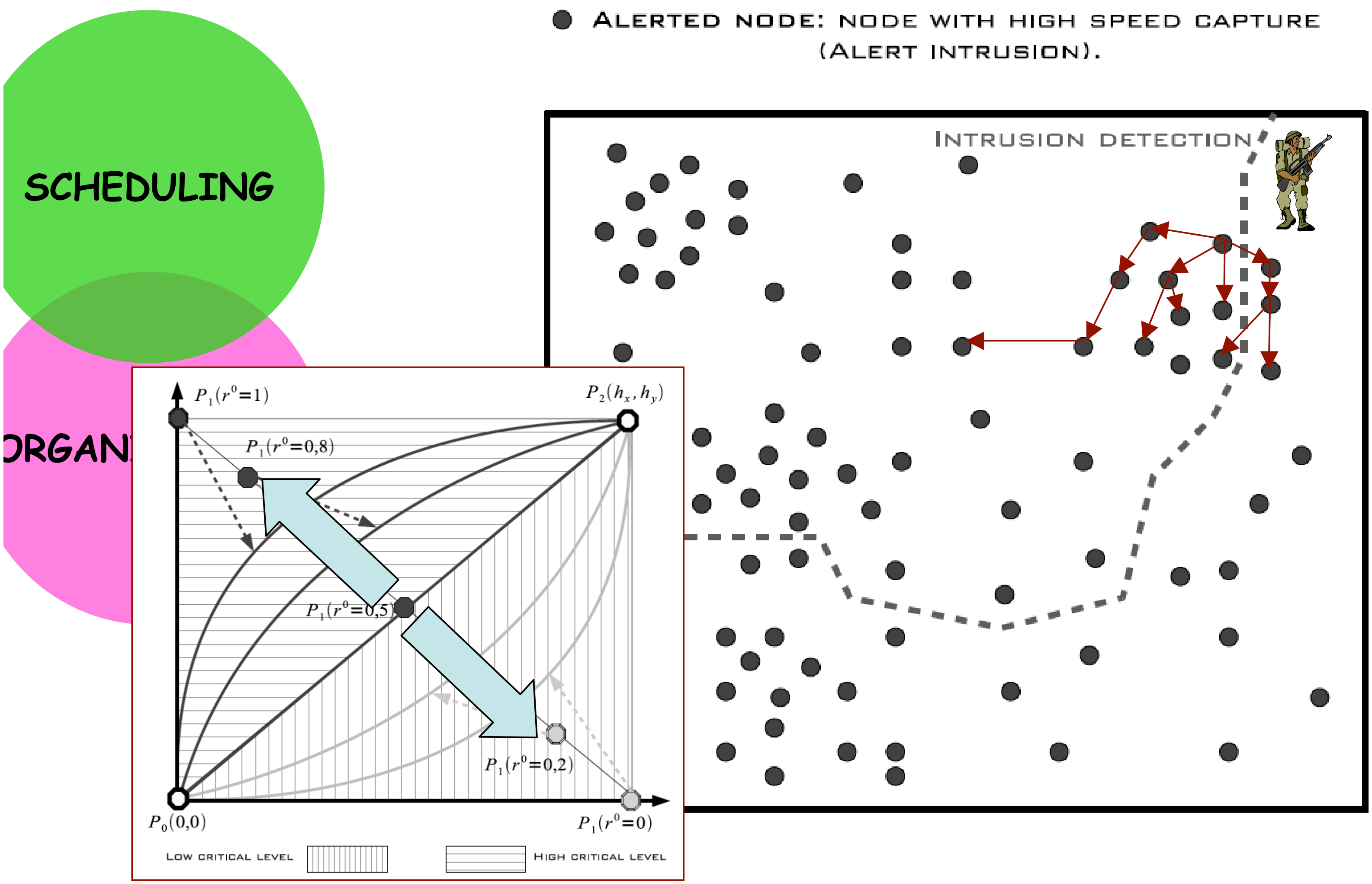


Controlled propagation (1)

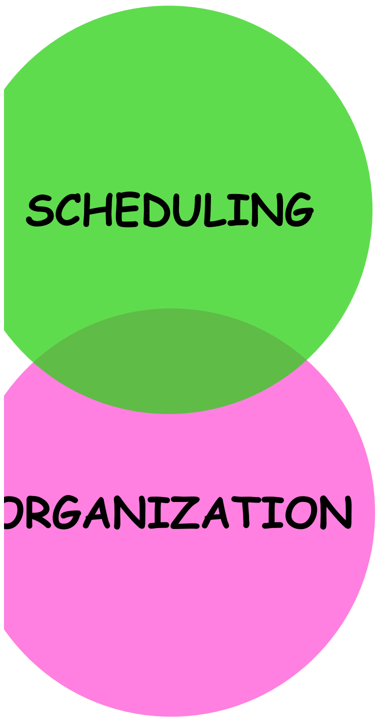




Controlled propagation (2)



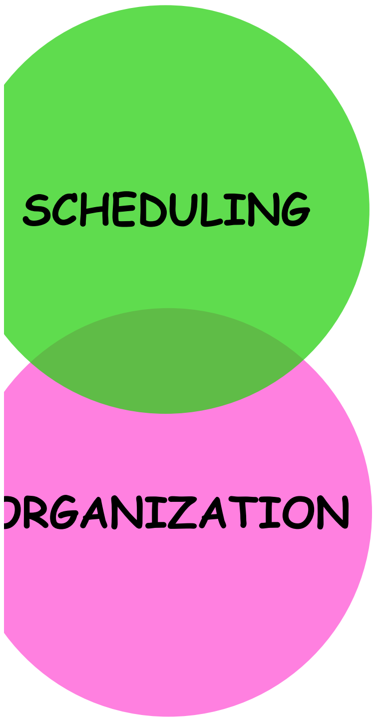
Controlled propagation (3)

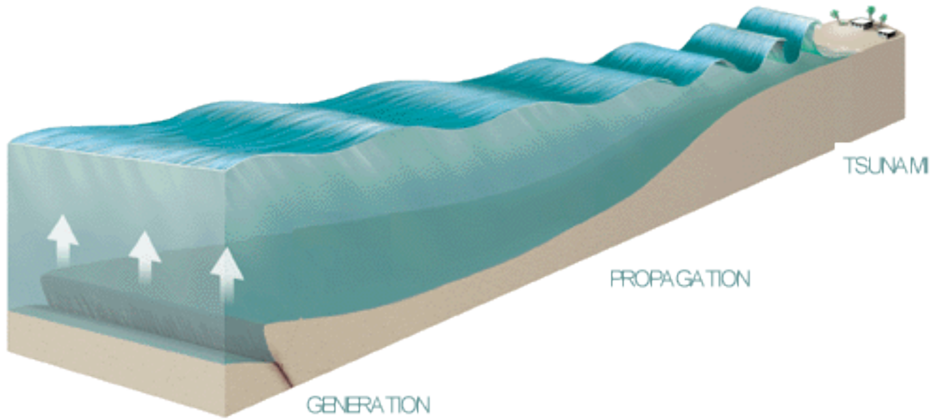


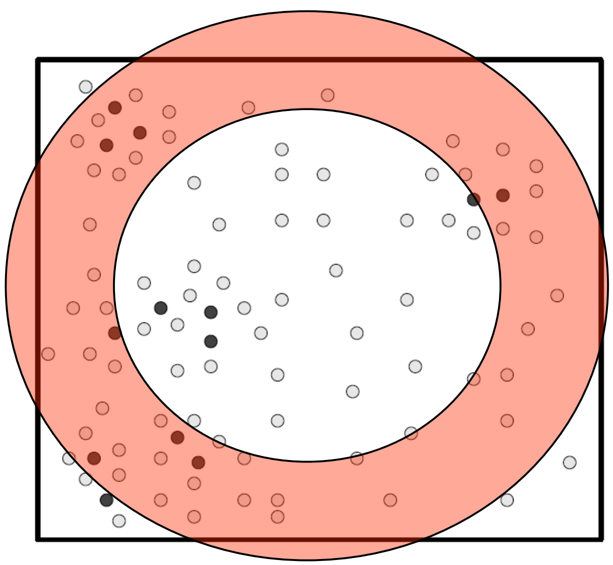
- Not a simple propagation or broadcast algorithm
  - Not all nodes need to be at the maximum (same) alert level
  - Which nodes should be more than others?
- Borrow propagation model from other disciplines
  - Epidemic propagation, percolation, wave propagation,…
  - According to the model, map the parameter of a surveillance system to the model's parameters

Controlled propagation (4)

ex: tsunami generation





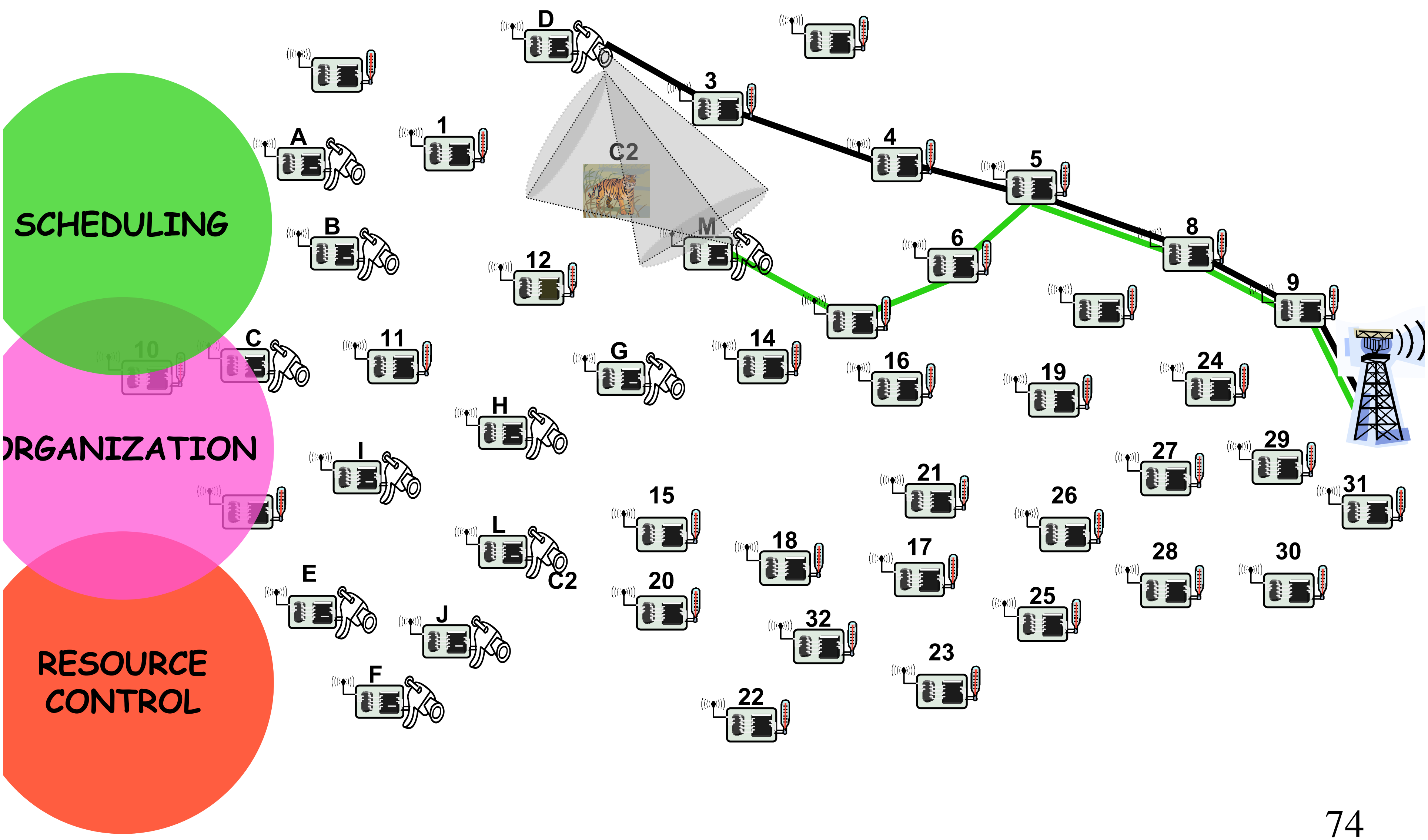


sensor nodes near the border may need to be « alerted » than others, they could have an amplification factor greater than those near the centre

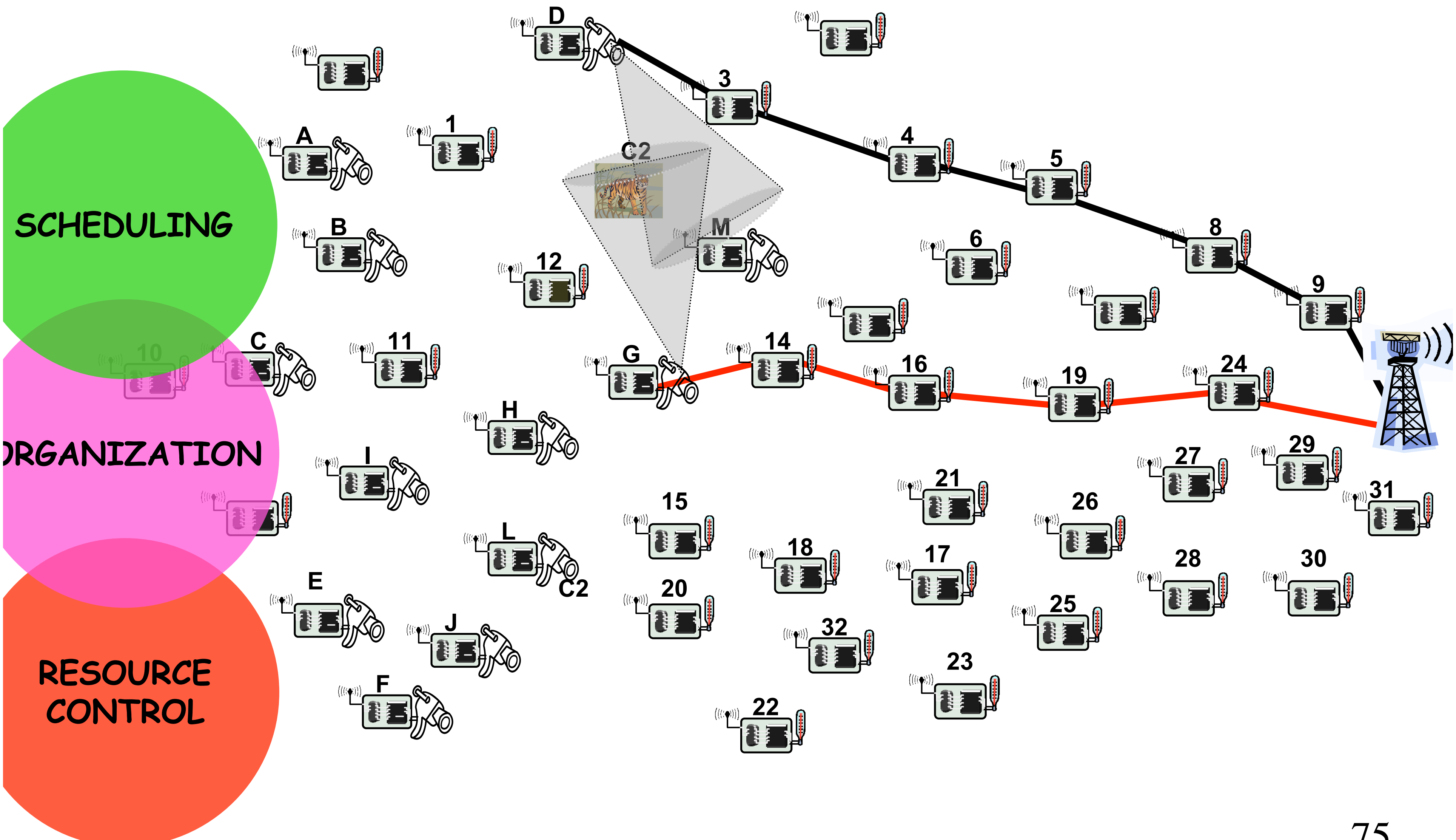
Congestion control (1)

- Lot's of sensor nodes=lot's of trafic
- High probability of bottleneck, lot's of packet drop, no useful data back to user!
- Scheduling is tighly linked to resource control to be efficient
- Scheduling is then not only find these nodes that « see » the event, but also how to select a subset of those nodes that minimizes congestion

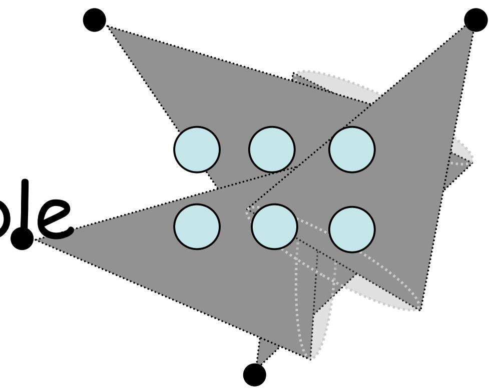
Congestion control (2)



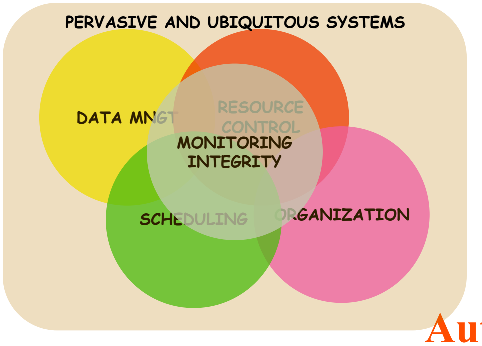
Congestion control (3)



Scheduling cover-set



- On intrusion, it is desirable to use more camera
  - To circumvent occlusions
  - To help for disambiguation
- It is not necessary that all activated camera capture at a same speed (probably high speed)
  - How define different the target frame capture speed for each node of the same cover set?



Towards wide-area situation awareness

Combination of randomly and manually deployed sensors