Optimal Path Evolution in a Dynamic Distributed MEMS-Based Conveyor

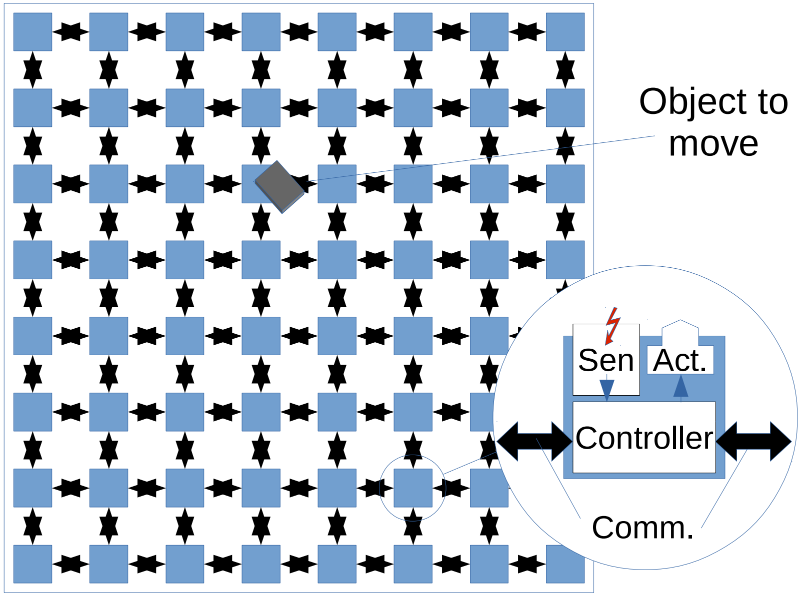
Haithem Skima, **Eugen Dedu**, Julien Bourgeois, Christophe Varnier, Kamal Medjaher

UBFC / UFC / ENSMM, Institut FEMTO-ST
France

DepCoS-RELCOMEX
Brunów, Poland
June–July 2016

The conveyor

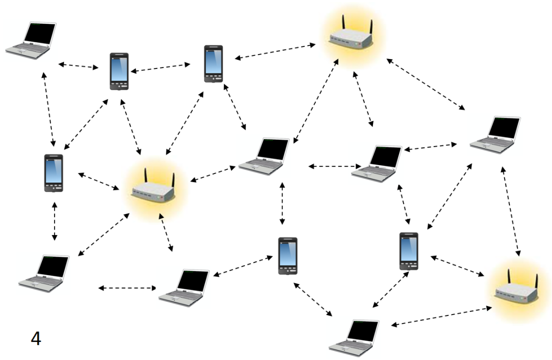
- A matrix of identical blocks
- Each block contains:
  - a sensor to detect if an object is above it
  - a controller
  - a power supplier
  - an actuator: MEMS valve (micro-electro-mechanical system)
  - a network module to communicate with its four immediate neighbours (up, down, left, right)
- The valve generates air, which allows to move the object above it
- The goal of the surface is to convey objects from one point to another



Context: intelligent transportation systems

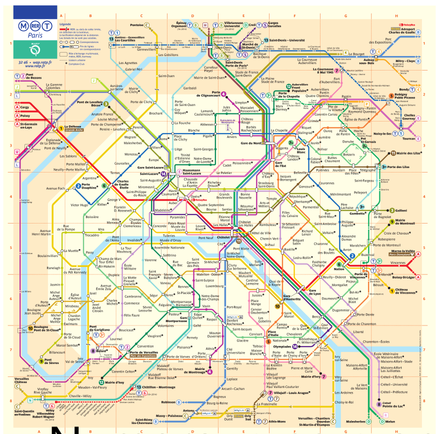
Common property: several possible paths for transportation

Network communication



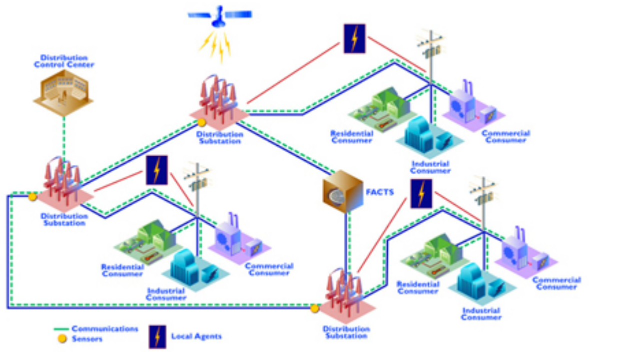
- Degradation upon usage (energy consumption)
- Goal: speed and dependability

Car traffic



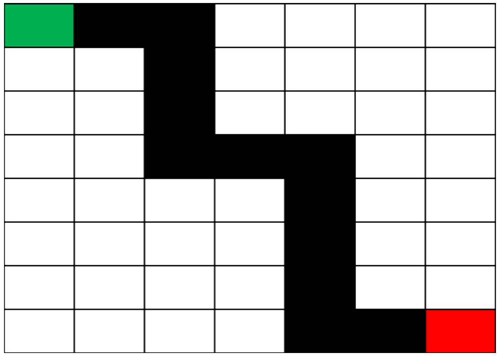
- Numerous cars on the same road
- Goal: decrease travel time

Energy network



- Cables limited to e.g. 100 MW
- Depend on time (peak hours)
- Goal: serve all clients

Conveyor



- Distributed and modular system
- Degradation upon usage (blocks degrade)
- All objects have the same destination
- One object at a time
- Goal: increase system's dependability (i.e. availability, reliability etc.) by carefully choose the transportation path
- **Conclusion:** each system has its specific features, leading to different properties; let's analyse conveyor dependability

Dependability issue

- One issue of the conveyor: blocks **degrade** (deteriorate) during usage, hence they can **fail**
- History of maintenance strategies:
  - unplanned breakdown strategy – *after* occurrence of failure
  - preventive maintenance – after some fixed time, regardless system's health
  - condition-based maintenance (CBM) – uses *current* system's health
  - recently, **predictive maintenance** (PM) – *predict* failure, system's remaining useful life (RUL), prognostics and health management (PHM) research field <== this is what we target

Degradation model used

- The degradation of a system is given by the degradation of its blocks, which in turn is given by the degradation of block components; we consider only MEMS valves
- During object conveyance, MEMS valves degrade
  - each time a MEMS is used to convey an object, its number of cycles C increments by 1
- Here we use a linear degradation:
  - RUL = initial_RUL – C
- Our later experiments yielded

  an exponential degradation:
  - health indicator = a.exp(b.C) + c.exp(d.C)

    with a,b,c,d constants
- Definitions:
  - path RUL = min (RUL) of all its blocks
  - optimal path = the path whose path RUL is max

S 95 – 1.5 (0,0)	98 – 1.2 (0,1)	57 – 5.3 (0,2)	23 – 8.7 (0,3)	64 – 4.6 (0,4)
14 – 9.6 (1,0)	44 – 6.6 (1,1)	16 – 9.4 (1,2)	88 – 2.2 (1,3)	58 – 5.2 (1,4)
79 – 3.1 (2,0)	83 – 2.7 (2,1)	27 – 8.3 (2,2)	83 – 2.7 (2,3)	22 – 8.8 (2,4)
44 – 6.6 (3,0)	98 – 1.2 (3,1)	72 – 3.8 (3,2)	96 – 1.4 (3,3)	99 – 1.1 (3,4) D

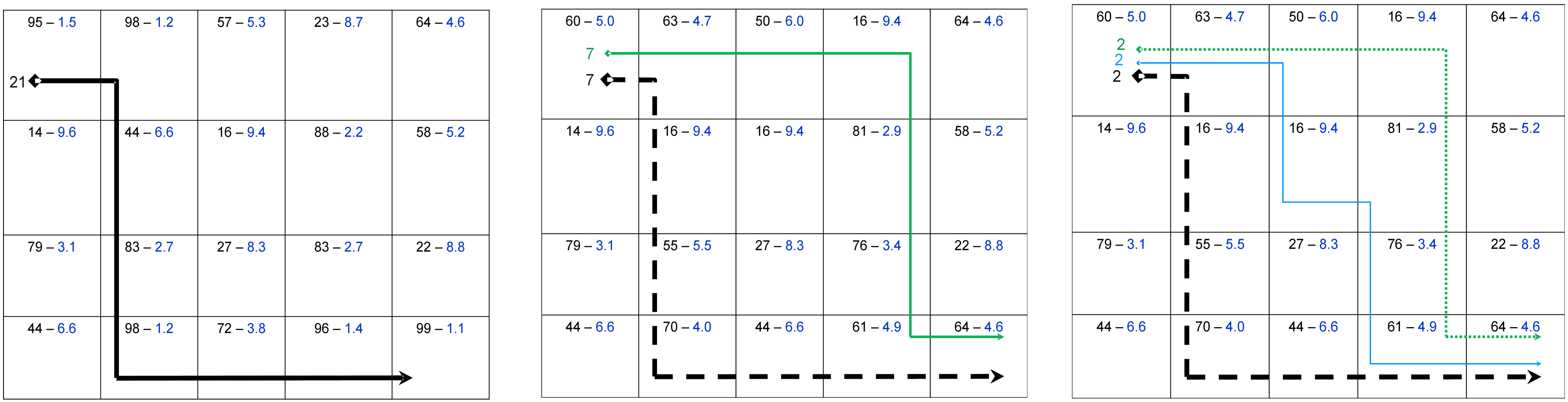
Simulator features

- Written in Java language, multi-threaded (each block is a thread)
- Allows to choose: surface dimension, number of objects to introduce, sources position
  - initial RULs are random
- During simulation, each time an object is conveyed: computation of the optimal path by each block (Dijkstra's algorithm), degradation of blocks, communication among blocks, statistics presentation (paths, surface state etc.)

Initial surface for simulations

95 – 1.5 (0,0)	98 – 1.2 (0,1)	57 – 5.3 (0,2)	23 – 8.7 (0,3)	64 – 4.6 (0,4)
14 – 9.6 (1,0)	44 – 6.6 (1,1)	16 – 9.4 (1,2)	88 – 2.2 (1,3)	58 – 5.2 (1,4)
79 – 3.1 (2,0)	83 – 2.7 (2,1)	27 – 8.3 (2,2)	83 – 2.7 (2,3)	22 – 8.8 (2,4)
44 – 6.6 (3,0)	98 – 1.2 (3,1)	72 – 3.8 (3,2)	96 – 1.4 (3,3)	99 – 1.1 (3,4)

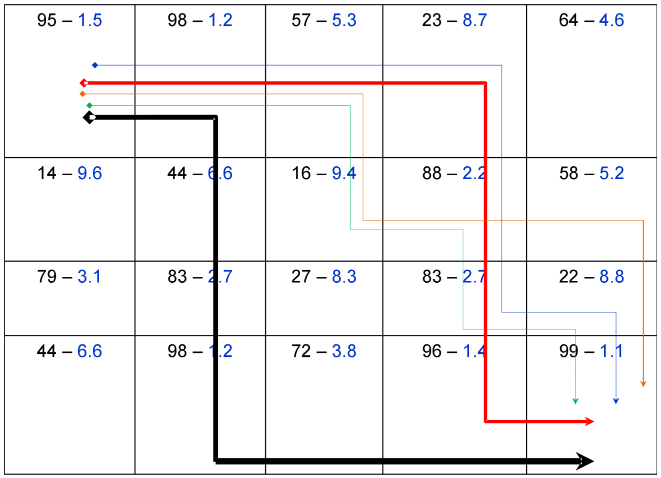
Optimal path evolution – one source

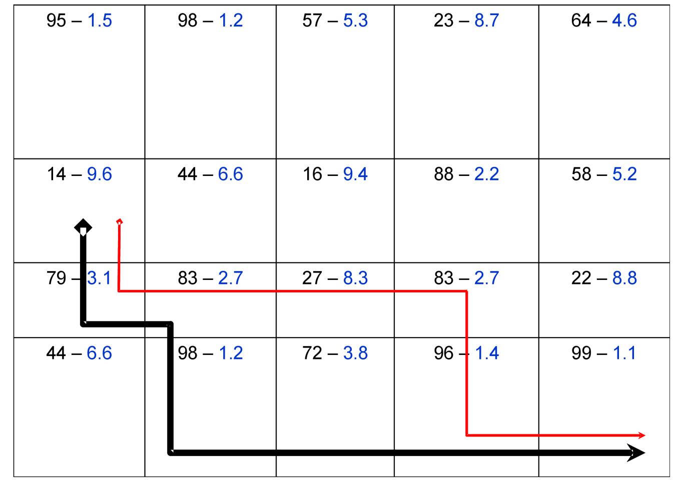


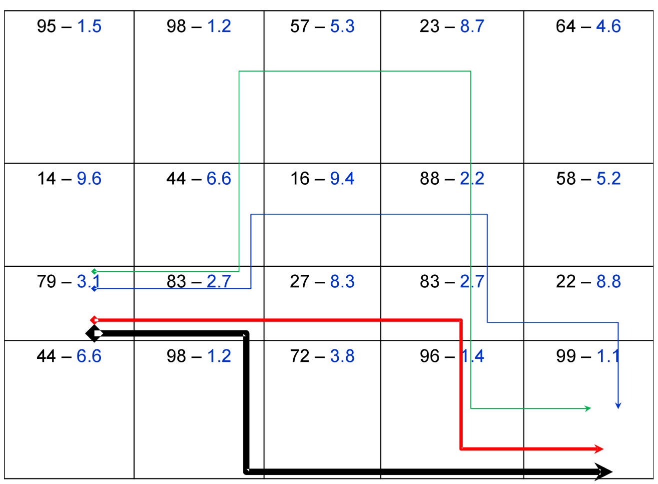
Path evolution:

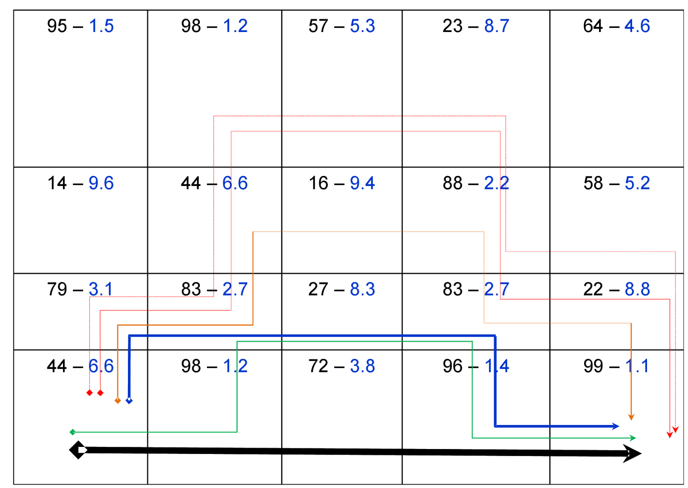
- Number of optimal paths:
  - a 1st optimal path, afterwards a new optimal path appears and 2 optimal paths are used, and so on
- Oscillation among optimal paths
- Duration of usage of each optimal path during one oscillation:
  - the more optimal paths, the less they oscillate
- Duration of usage of each optimal path during the whole simulation:
  - the 1st optimal path is the most used, afterwards the 2nd optimal path etc.

Two and more sources









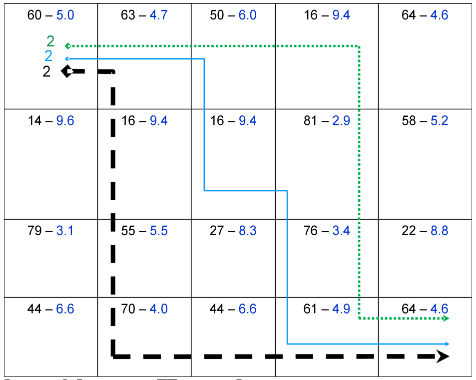
Conclusions:

- For two sources, paths oscillate more or less randomly
- For four and more sources, optimal paths change randomly (no oscillation noticed)

Additional results found in the article

- In addition to the RUL, each block has a transfer time too, which is used in case of two equal optimal paths (i.e. with the same RUL)
- It is better to use RUL, not transfer time, as main criterion for paths
- Position of the sources influences surface lifetime

Conclusions and perspectives



- The number of optimal paths increases with time
- Optimal paths oscillate
- The more optimal paths, the less they oscillate
- The 1st optimal path gets used the most often, afterwards the 2nd optimal path and so on
- For several sources, there is no pattern anymore
- Perspectives:
  - allow several objects concurrently on the surface
  - use a more realistic movement model, by taking into account path curves in object speed, object inertia, more realistic degradation model etc.
- Simulator code used to get results is available at http://eugen.dedu.free.fr