Yard traffic and congestion in container terminals

Ilaria Vacca
Transport and Mobility Laboratory, EPFL

joint work with
Michel Bierlaire, Matteo Salani & Arnaud Vandaele

6th International Conference on Computational Management Science

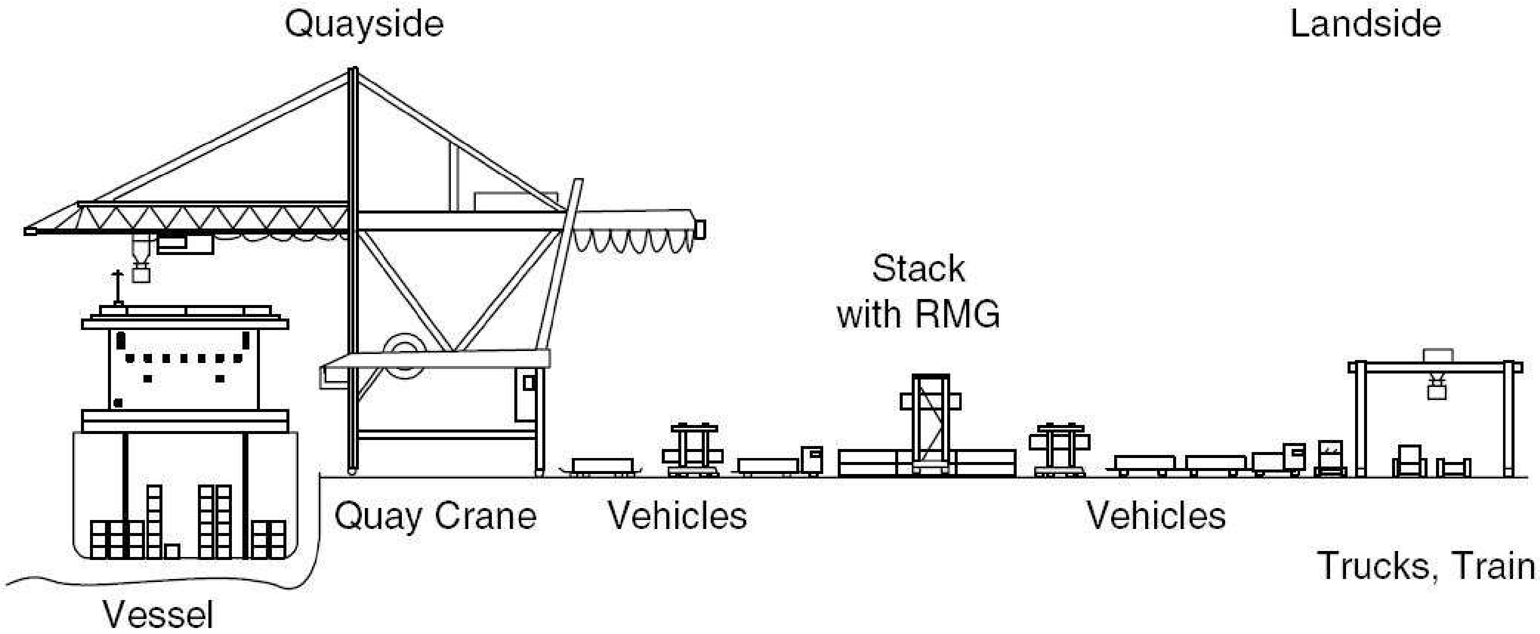
May 1st, 2009

Outline

- Introduction and motivation
- Modeling
- Congestion measures
- Optimization
- Computational results
- Future work

Container Terminals (CT)

- Zone in a port to import/export/transship containers
- Different areas in a terminal: berths, yard, gates
- Different types of vehicles to travel between the yard and the berth



Motivation

- Along the quay, containers are loaded/unloaded onto/from several boats
- Containers' transfer lead to a high traffic in the yard zone

- The **berth&yard allocation plan** assigns ships to berths and containers to yard blocks
- Terminal planners usually minimize the total distance travelled by the carriers, disregarding:
    - Congestion issues (operations slowdowns because of bottlenecks)
    - Alternative solutions (symmetries)

Aim of this study:

- ✓ Model the terminal and develop measures of congestion
- ✓ Evaluate the impact of the optimization of such measures on the terminal

Assumptions

- We take into account flows of containers from the quayside to the yard
- Given a berth&yard allocation plan, we define a **path** as an OD pair:
  - origin (berth)
  - destination (block)
  - number of containers
- We consider flows of containers over a working shift
- Decisions could be taken on:
  - the berth allocation plan (berths and ships)
  - the yard allocation plan (destination blocks)
  - demand splitting over blocks

➔ In this study: given a set of *p* paths, determine the destination blocks

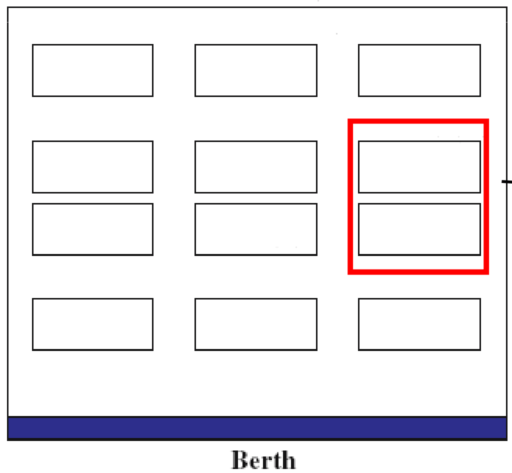
Literature

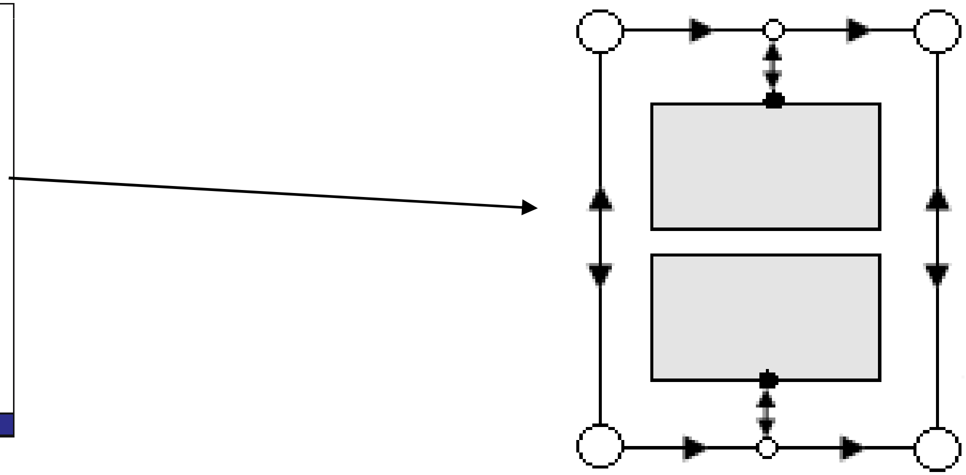
- Layout:
  - Kim et al. **An optimal layout of container yards**, OR Spectrum, 2007.

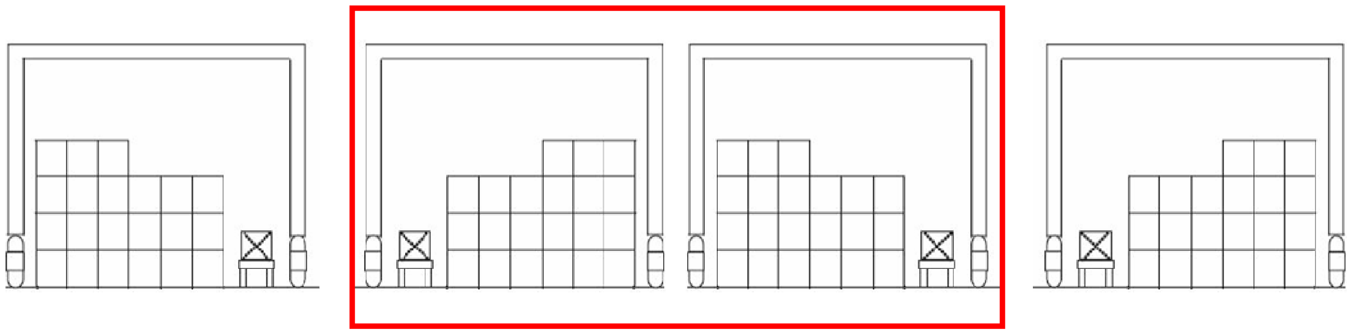
- Congestion:
  - Lee et al. **An optimization model for storage yard management in transshipment hubs**, OR Spectrum, 2006.
  - Beamon. **System reliability and congestion in a material handling system**, Computers Industrial Engineering, 1999.

Modeling the terminal

Basic element

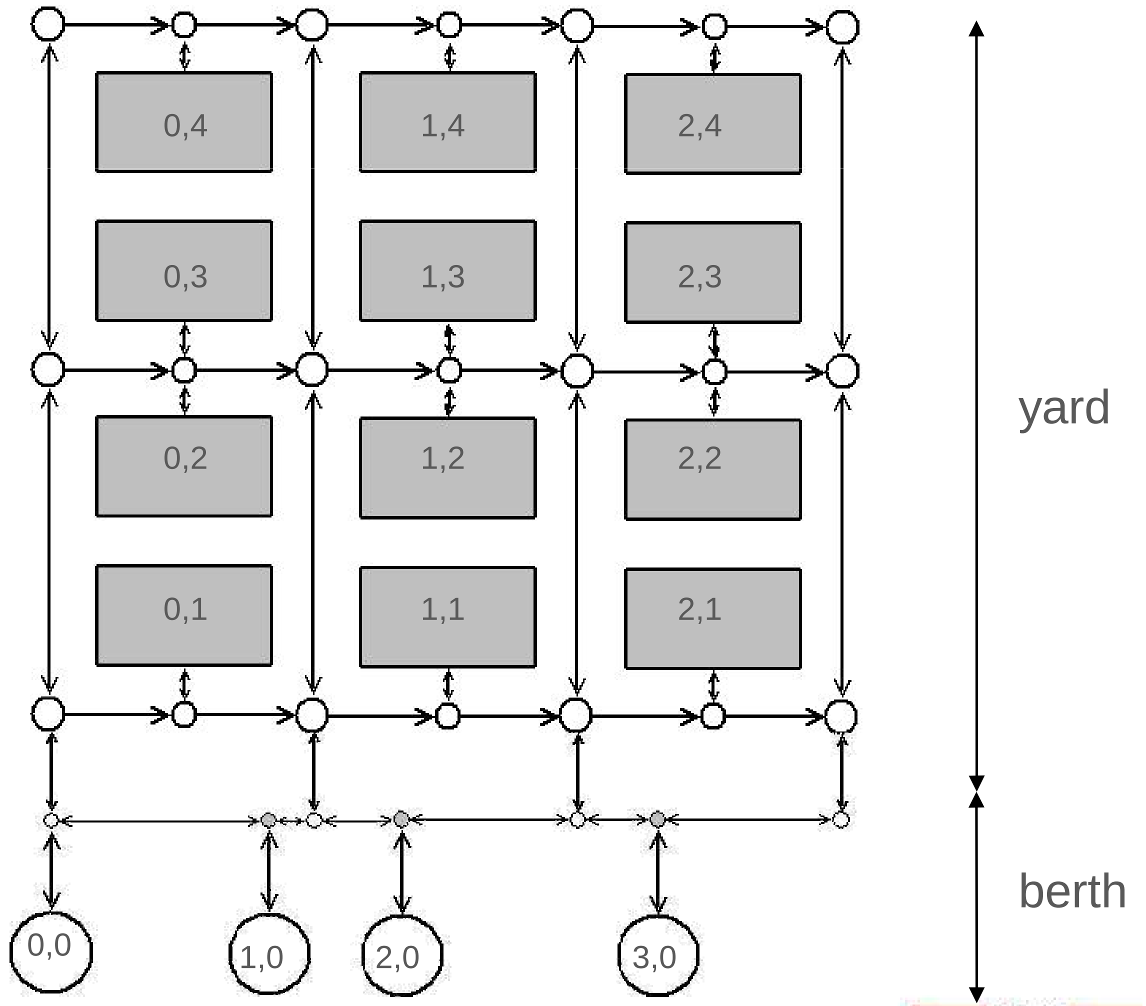






Modeling the terminal

- ***(m* x *n)*** basic elements of 2 blocks each compose the yard
- coordinates system for OD pairs ***$(x_o, y_o) – (x_d, y_d)$***
- only berth-to-yard and yard-to-berth paths are considered



TRANSP-OR

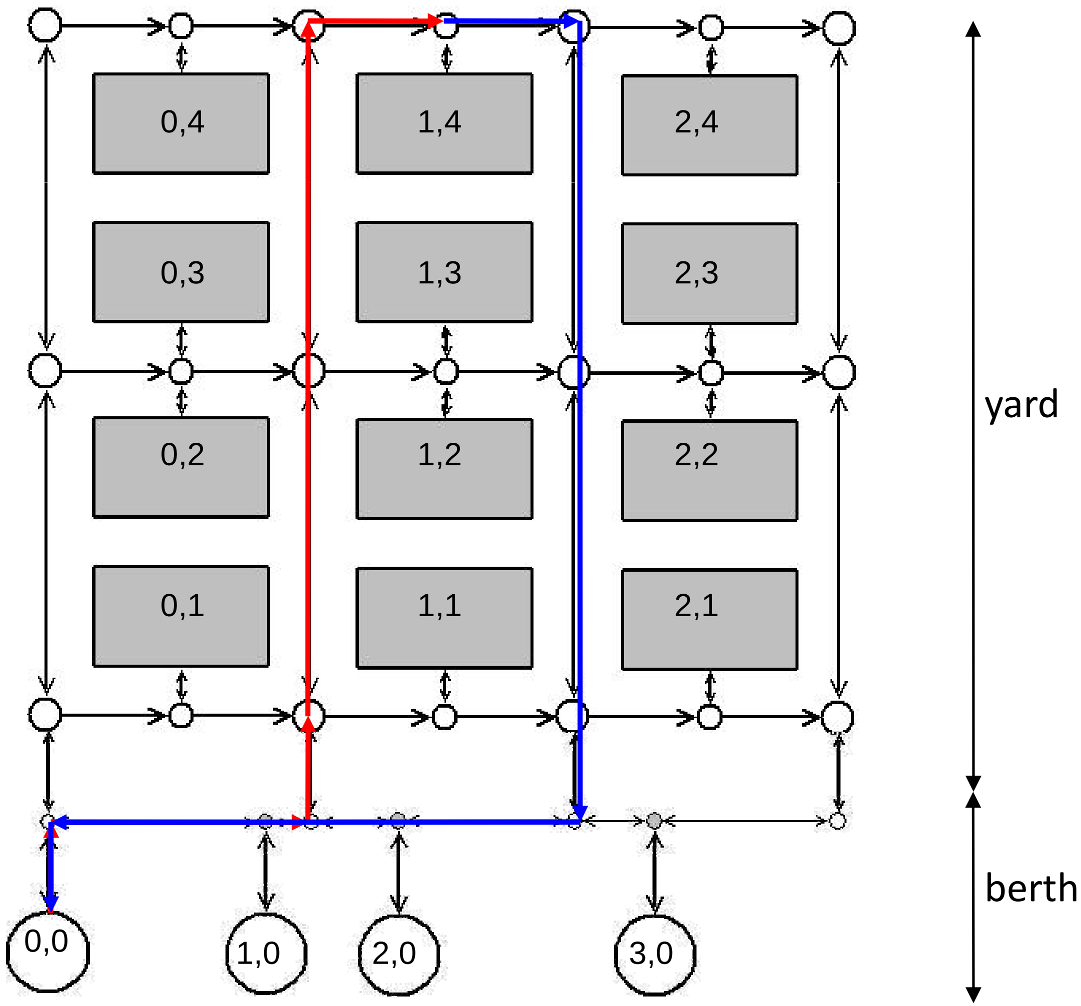
ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Routing rules

- Horizontal lanes are one way
- Vertical lanes are two way

- Toward the block, closest left vertical lane, turn right.
- Toward the quay, turn right at the first vertical lane.
- Back to origin berth position.

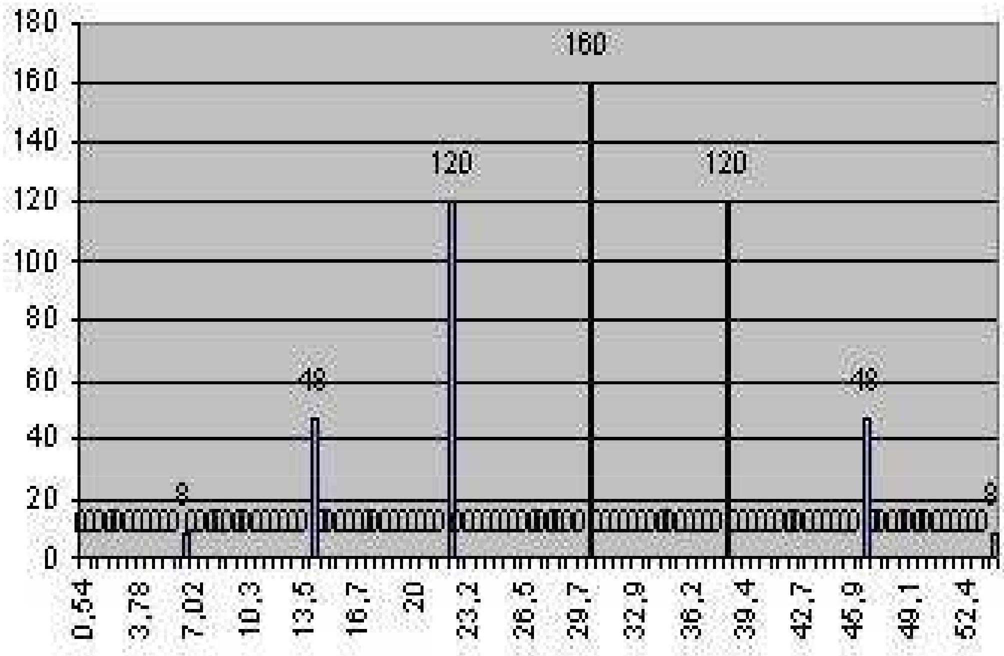
- Distance travelled, closed formula (Manhattan)



Symmetries

Minimize distance:

in a 2x2 yard with 2 paths, no capacity on blocks



Number of solutions with equal distance

Congestion measures

- **Aim of the study:**
  - estimate the state/congestion of a yard when implementing a plan
  - provide simple closed formulas, to be used as secondary objectives

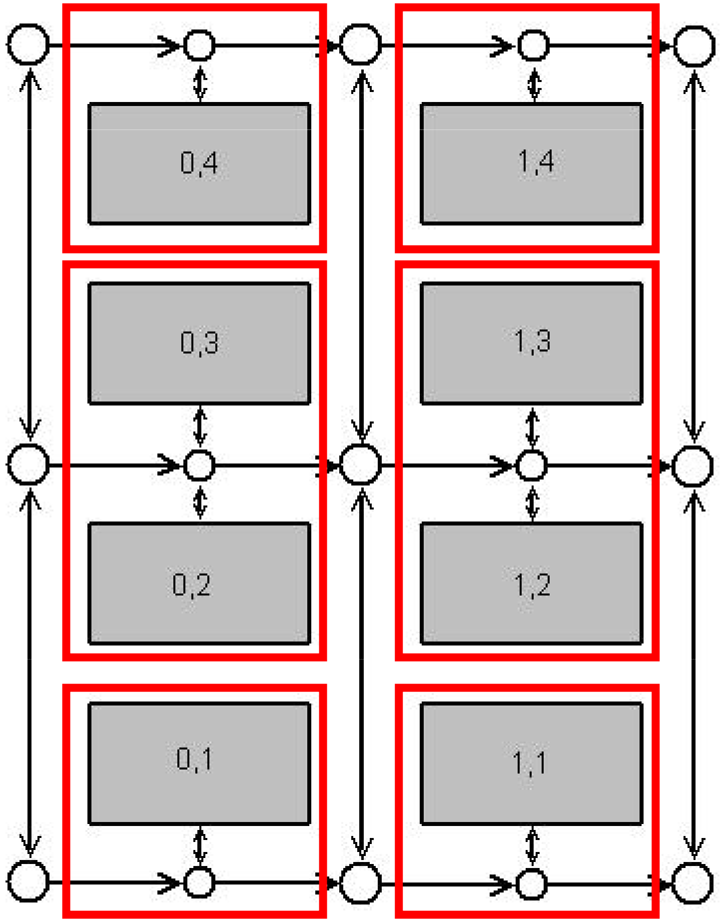
- **Factors taken into account:**
  - interference between blocks sharing the same lane
  - lane congestion
  - interference between paths

1. Block congestion

- **congestion among blocks sharing the same lane**
- **“area”:** blocks with the same entrance node
  - # of areas: ***s = 2n + n(m-1)***
  - c_j : *# of containers on path j = 1...p*
  - N_i : # of containers allocated to area *i*
  - ***N**** : # of containers in each area in the optimal solution (even distribution among areas)

$$C_b = \frac{D}{D_{max}} = \frac{\sum_{i=1}^{s} |N_i - N^*|}{\frac{2(s-1)}{s} \sum_{j=1}^{p} c_j}$$

- 1-norm and 2-norm w.r.t. the best over the worst case

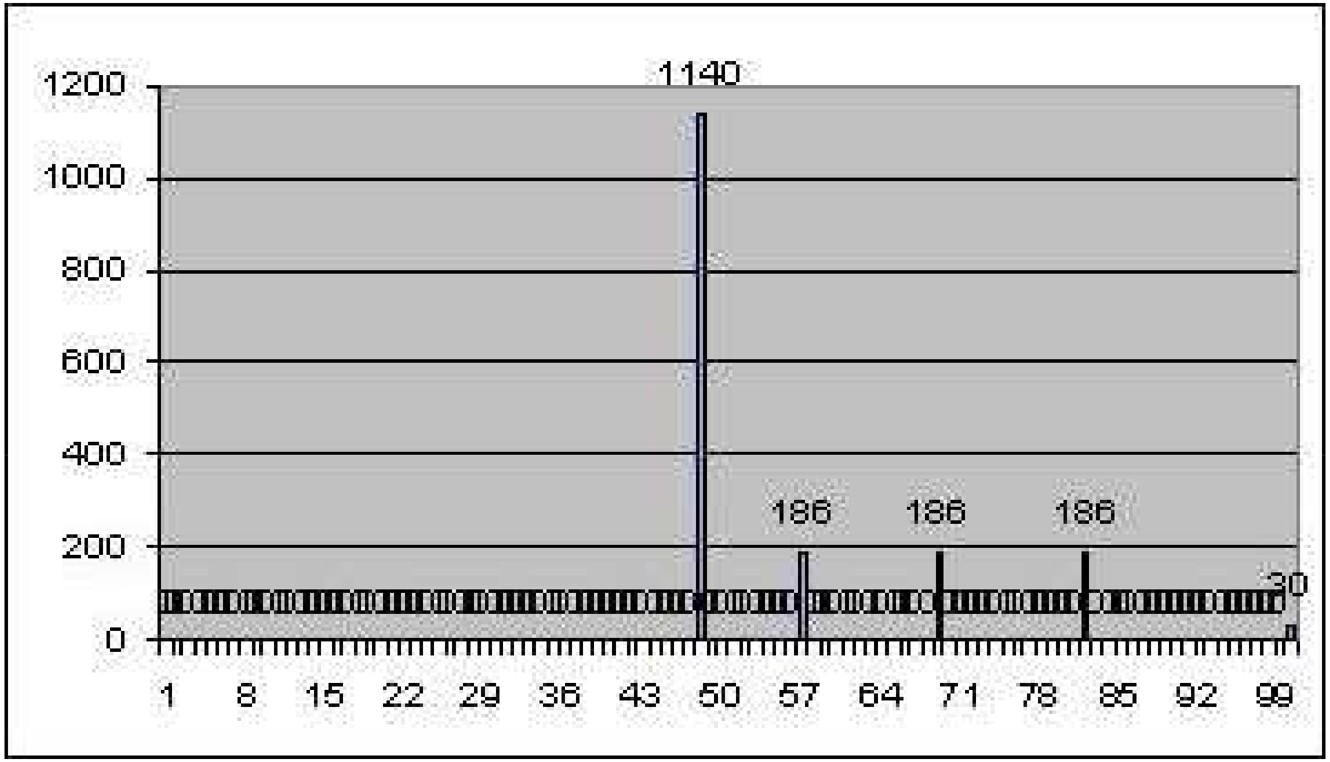


TRANSP-OR

EPFL ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

1. Block congestion

- 3 paths in a 2x3 yard (12 blocks) → possible solutions : $12^3 = 1728$
- number of solutions with same block congestion (distribution of 2-norm C_b) :



TRANSP-OR

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

2. Edge congestion

- this indicator simply measures the average traffic over an edge

$$\theta = max_k f_k \qquad \theta_{max} = \sum_{j=1}^{p} c_j \qquad \mu_{min} = 0$$
$$\mu = min_k f_k$$

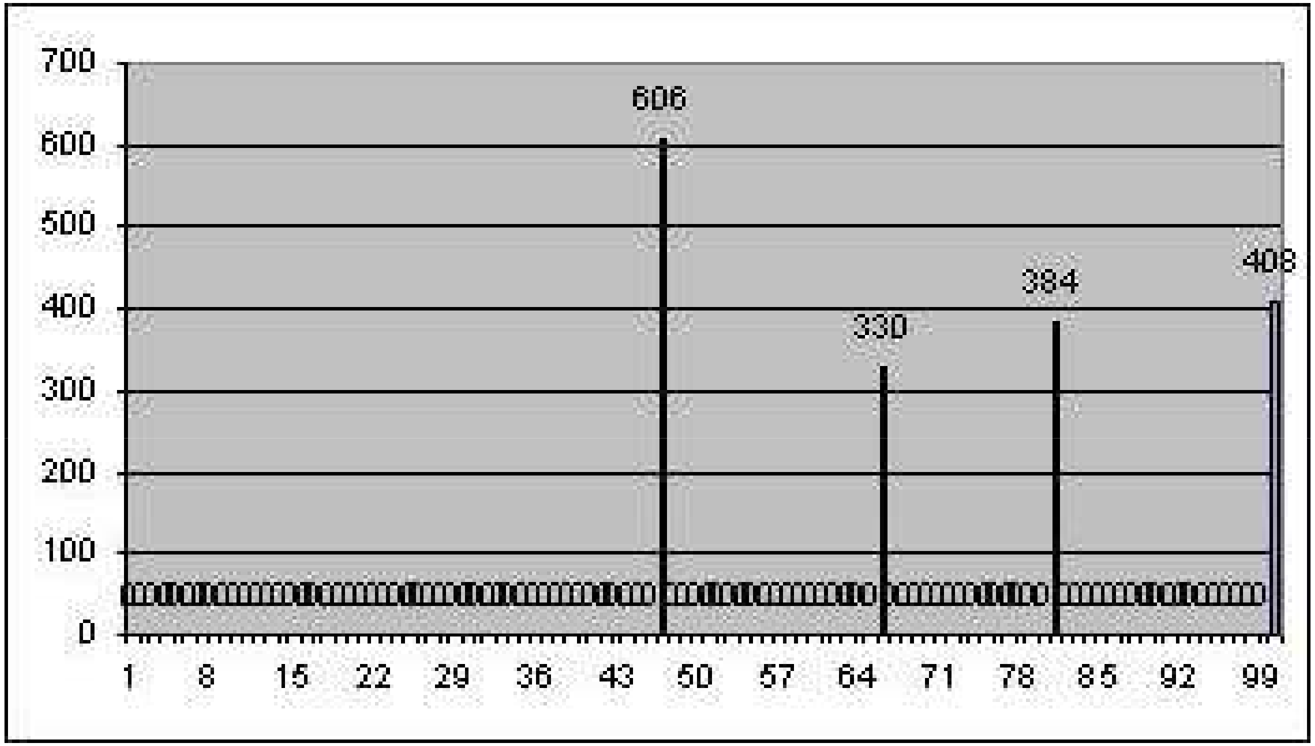
$$C_e = \frac{\theta - \mu}{\theta_{max} - \mu_{min}} = \frac{\theta - \mu}{\sum_{j=1}^{p} c_j}$$

- the best traffic situation is when flows are spread over the network: $\mu^* = \frac{\sum_{j=1}^{p} c_j}{n}$

$$C_e = \frac{\theta - \mu^*}{\theta_{max} - \mu^*} = \frac{\theta - \mu^*}{\sum_{j=1}^{p} c_j - \frac{\sum_{j=1}^{p} c_j}{n}} = \frac{(n)\theta - \sum_{j=1}^{p} c_j}{(n-1)\sum_{j=1}^{p} c_j}$$

2. Edge congestion

- 3 paths in a 2x3 yard (12 blocks) → possible solutions : $12^3 = 1728$
- number of solutions with same edge congestion (distribution of improved C_e):

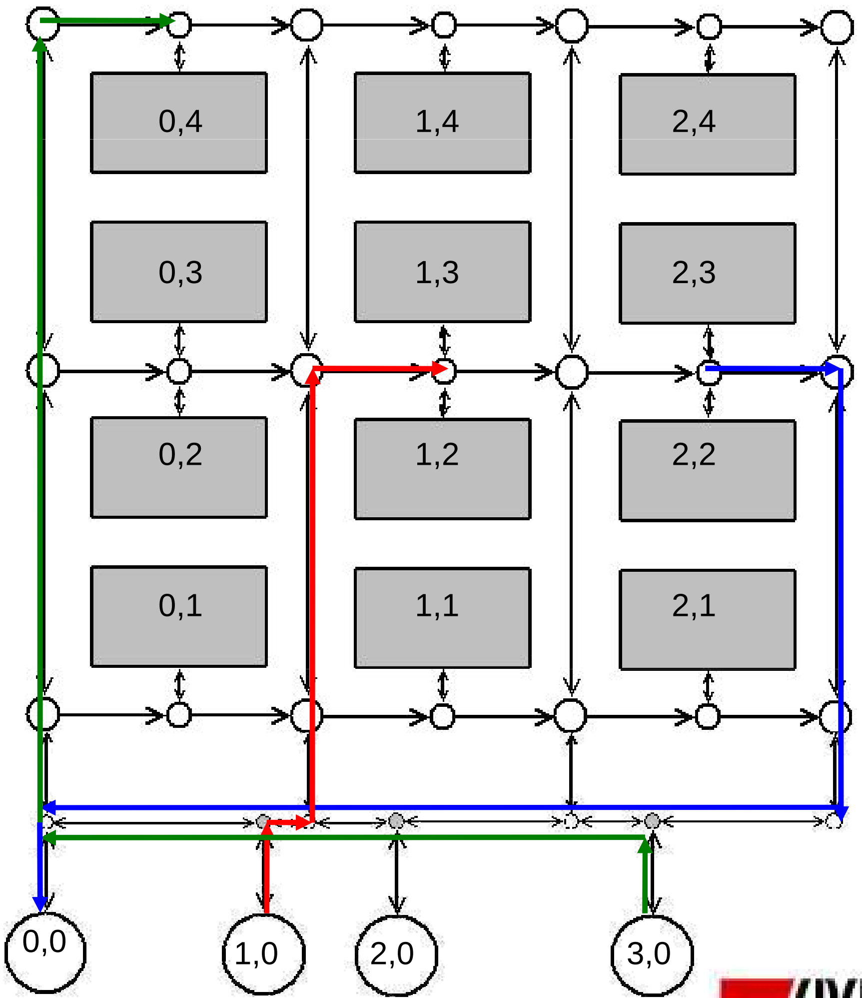


3. Path congestion

- **interference among “crossing” paths**

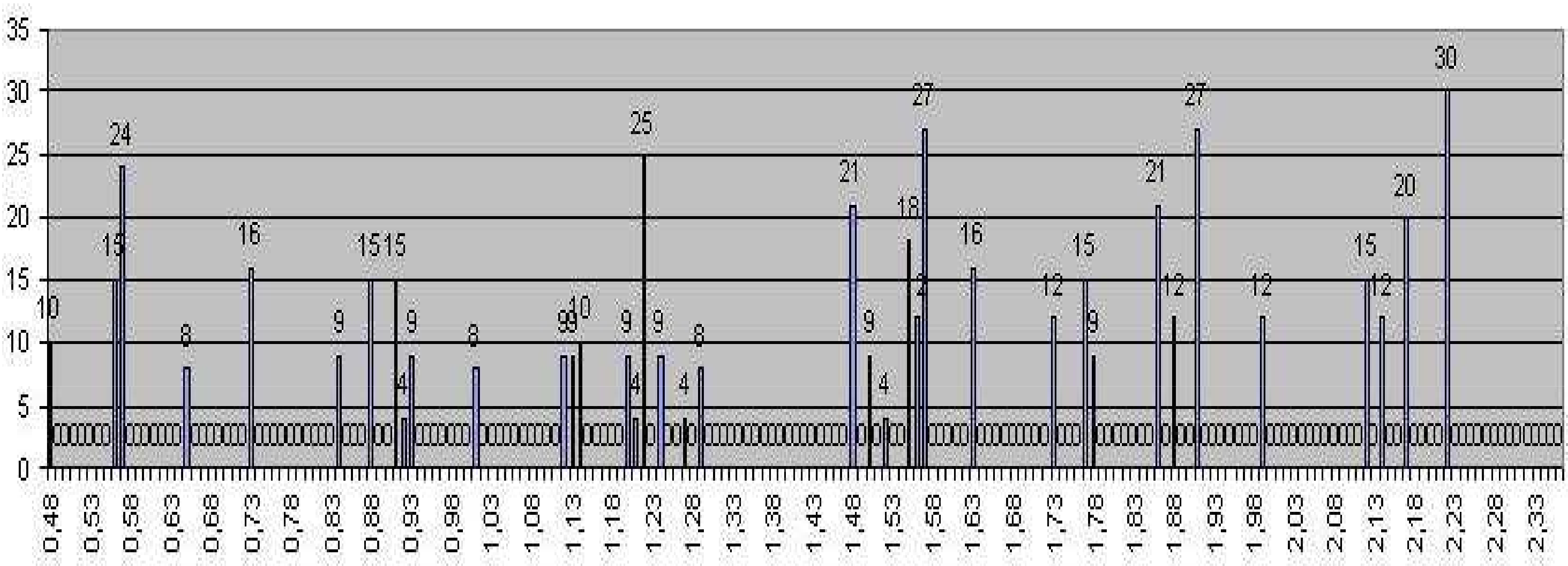
  - proximity matrix ***P*** *(2p X 2p)*
  - *p* berth-to-yard + *p* yard-to-berth paths
  - *P* is symmetric, 0 on the diagonal, 1 if two paths are “neighbours”
  - definition of *P* is influenced by routing rules
  - worst case: all 1 matrix (except diagonal)

$$C_p = \frac{p}{N_{max}} = \frac{1^T.P.c}{(2n-1)\sum_{i=1}^{2n} c_i}$$



Example

- 3 paths in a 2x3 yard
- Distribution of the objective function $z = \lambda_b . C_b + \lambda_e . C_e + \lambda_p . C_p$



Example

Objective function : $z = \lambda_b.C_b + \lambda_e.C_e + \lambda_p.C_p$

	Nb solutions	Nb different values	MIN	Nb MIN	CPU (s)
(2x2) – 3 paths	512	46	0,4764	10	0,2
(2x2) – 4 paths	4096	282	0,3473	30	1,4
(2x2) – 5 paths	32768	1831	0,5068	21	12,23
(2x2) – 6 paths	262144	7354	0,461	12	112,85
(2x3) – 3 paths	1728	52	0,4764	116	0,67
(2x3) – 4 paths	20736	470	0,3473	350	7,29
(2x3) – 5 paths	248832	4271	0,13	108	121,65

Optimization algorithm: GRASP

- GRASP: Greedy Randomized Adaptive Search Procedure

- Objective: assign a destination to each path such that congestion is minimized

- The algorithm builds a solution iteratively:
    - at each step, the destination for one specific path is chosen

Optimization algorithm: GRASP

	MIN	CPU (s) (enumeration)	CPU (s) (algorithm)	Nb iteration to reach optimum
(2x2) – 3 paths	0,4764	0,2	0,1	5
(2x2) – 4 paths	0,3473	1,4	0,2	10
(2x2) – 5 paths	0,5068	12,23	0,5	30
(2x2) – 6 paths	0,461	112,85	3	150
(2x3) – 3 paths	0,4764	0,67	0,1	5
(2x3) – 4 paths	0,3473	7,29	0,1	5
(2x3) – 5 paths	0,13	121,65	0,5	25
(2x3) – 6 paths	0,1953	??	15	1000

Computational tests

More realistic instances

	in 0,1s	in 1s	in 5s	in 10s	in 20s	in 60s
(3x10) – 3	0,4764	0,4764	0,4764	0,4764	0,4764	
(3x10) – 4	0,3473	0,3473	0,3473	0,3473	0,3473	
(3x10) – 5	0,13	0,13	0,13	0,13	0,13	
(3x10) – 6	0,389	0,195	0,195	0,195	0,195	
(3x10) – 7	0,343	0,267	0,267	0,267	0,267	
(3x10) – 8	0,26	0,1692	0,1646	0,1646	0,1646	
(3x10) – 9	0,304	0,2763	0,2763	0,2763	0,2763	
(3x10) – 15	0,2446	0,1931	0,1705	0,1582	0,1817	0,1602
(3x10) – 20	0,3275	0,2276	0,1663	0,1624	0,1609	0,1389

Conclusions and Outlook

- simple closed formulas to evaluate congestion in container terminals
- useful to differentiate symmetric solutions with equal distance

Ongoing work:

- validation of our approach via a CT simulator
- multi-objective optimization problem (explore other than weighted sum)
- improve the algorithm: study an exact approach; relax the assumptions, i.e. extend the set of possible decisions (berth allocation, demand splitting)

Thanks for your attention!