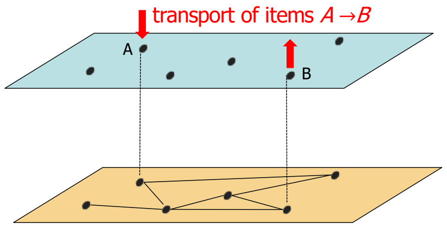
Inverse Problems in Network Science

Piet Van Mieghem

University of Electronic Science and Technology of China
16 May 2023, Chengdu

Network = Process + Graph



Process (Function)

Software, service

Graph (Structure)

Hardware, Topologie
Relationship between items and/or processes

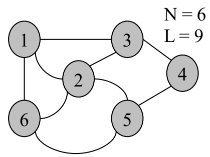
Network Science: Theory of processes on/in graphs

⬇

Duality between **process** and **graph** is cornerstone

Three equivalent representations of an undirected graph

Topology domain



$$A_{N\times N} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

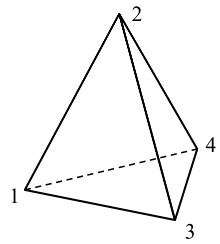
Spectral domain

$$A = A^T = X\Lambda X^T$$

$X_{N\times N}$: orthogonal eigenvector matrix

$\Lambda_{N\times N}$: diagonal eigenvalue matrix

Geometric domain

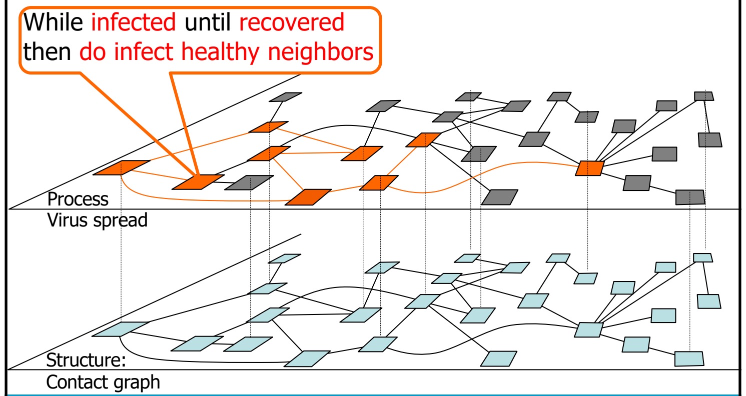


Each undirected graph with N nodes = a simplex in Euclidean *(N-1)*-dimensional space

Devriendt, K. and P. Van Mieghem, "The Simplex Geometry of Graphs", Journal of Complex Networks, Volume 7, Issue 4, pp. 469–49 August 2019. (http://arxiv.org/abs/1807.06475).

3

Local Rule – Global Emergent behavior

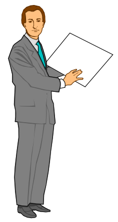


LRGE dynamics:

$$\frac{dx_i(t)}{dt} = f_i(x_i(t)) + \sum_{j=1}^{N} a_{ij}\, g(x_i(t), x_j(t))$$

4

Outline



Inverse **linear** process:

Inverse all shortest path problem

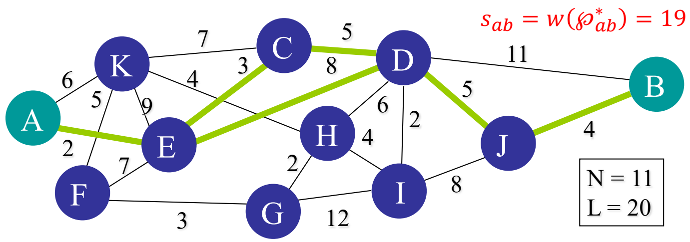
Inverse **non-linear** process:

Prediction of LRGE dynamics

TUDelft

5

Shortest Path Routing



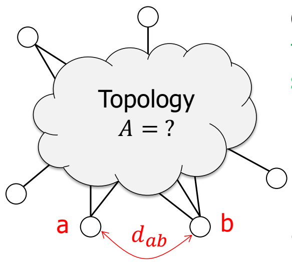
- Link metric is non-negative and *additive* (e.g. delay, cost,...)
- Weight of path $\wp_{ij}$ is $w(\wp_{ij}) = \sum_{l \in \wp_{ij}} w_l$
- The shortest path $\wp^*_{ij}$ is minimizer of $s_{ij} = w(\wp^*_{ij}) \le w(\wp_{ij})$
- S: shortest path weight matrix with elements $s_{ij} = w(\wp^*_{ij})$

P. Van Mieghem, *Data Communications Networking*, (ISBN: 978-94-91075-01-8), 2011

TUDelft

6

Inverse All Shortest Path Problem (IASPP)



Given an $N \times N$ demand matrix D
find the *weighted* adjacency matrix A
subject to

1) each shortest path weight obeys

$$s_{ab} \leq d_{ab}$$

for any node pair (a,b) in the graph with N nodes

2) $\|D - S\|$ is minimized

Motivation:

- end-to-end delay (QoS) in telecommunications
- travel times in transportation
- seismic tomograph (earthquakes) of geologic zones
- EEG/MEG in human brain

7

$$s_{ab} = w(\wp^*_{ab}) = \sum_{l \in \wp^*_{ab}} w_l$$

D: not necessarily a distance matrix
S: always a distance matrix

7

Function of network

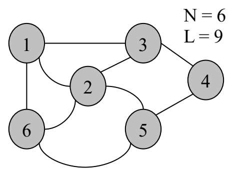
- Usually the function of a network is related to the *transport of items over its underlying graph*
- In man-made infrastructures: two major types of transport
  - Item is a flow (e.g. electrical current, water, gas,…)
  - Item is a packet (e.g. IP packet, car, container, postal letter,…)
- Flow equations and physical laws determine transport
  Maxwell equations, Kirchhoff & Ohm, hydrodynamics, Navier-Stokes equation (turbulent, laminar flow equations, etc.), epidemic spread, ...
- Protocols determine transport of packets (IP protocols and IETF standards, car traffic rules, etc.)

8

TU Delft

8

Laplacian matrix Q



$$Q_{N\times N} = \begin{bmatrix} 3 & -1 & -1 & 0 & 0 & -1 \\ -1 & 4 & -1 & 0 & -1 & -1 \\ -1 & -1 & 3 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & -1 & 0 & -1 & 3 & -1 \\ -1 & -1 & 0 & 0 & -1 & 3 \end{bmatrix}$$

Degree of node $d_i = \sum_{k=1}^{N} a_{ki}$

$Q = \Delta - A$ A : adjacency matrix and $\Delta = diag(d_1\ d_2 \cdots d_N)$

9

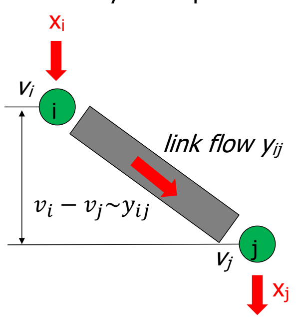
P. Van Mieghem, Graph Spectra of Complex Networks, 2nd edition, Cambridge University Press, 2023 (to appear)

TU Delft

9

Linear dynamics on flow networks

Linear dynamic process: "proportional to" (~) graph of network



Examples:
- water (or gas) flow ~ pressure
- displacement (in spring) ~ force
- heat flow ~ temperature
- electrical current ~ voltage

x	=	Q	.	v
injected nodal current vector		weighted Laplacian of the graph		nodal potential vector

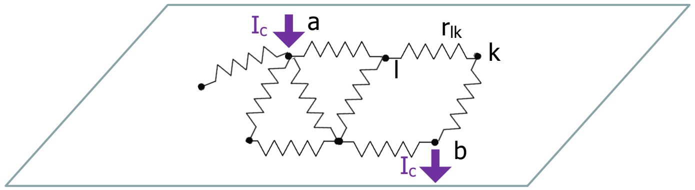
Inverse of $x = Qv$ is $v = Q^{\dagger}x$ subject to $u^T v = 0$ (average potential is zero)
$Q^{\dagger}$ is the pseudoinverse of the Laplacian matrix

10

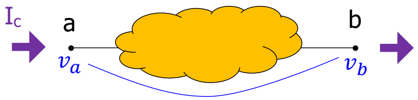
P. Van Mieghem, K. Devriendt and H. Cetinay, 2017, "Pseudoinverse of the Laplacian and best spreader node in a network", Physical Review E, vol. 96, No. 3, p 032311.

TU Delft

10



effective resistance ω_{ab} ⇕ $\omega_{ab} = Q^{\dagger}_{aa} + Q^{\dagger}_{bb} - 2Q^{\dagger}_{ab}$



$$v_a - v_b = I_c \omega_{ab}$$

Path networks are more confined than flow networks: $\omega_{ij} \le s_{ij}$

The effective resistance matrix Ω is the **flow analogon** of the shortest path weight matrix S

TUDelft
Technische Universiteit Delft

11

The famous Fiedler's block inverse

$$\begin{bmatrix} 0 & u^T \\ u & \Omega \end{bmatrix}^{-1} = \begin{bmatrix} -2\sigma^2 & p^T \\ p & -\frac{1}{2}Q \end{bmatrix} \qquad \Omega p = 2\sigma^2 u$$

u: all-one vector

there the Laplacian $Q = \Delta - A$ and $\Delta = diag(d)$

Applying inverse block matrix formulae,

$$p = \frac{1}{u^T\Omega^{-1}u}\Omega^{-1}u \qquad 2\sigma^2 = \frac{1}{u^T\Omega^{-1}u}$$

$$A = \Delta + 2\Omega^{-1} - \frac{1}{\sigma^2}pp^T$$

Inverse **flow** problem is exactly solvable!

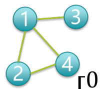
12

Van Mieghem, P., 2021, "A tree realization of a distance matrix: the inverse shortest path problem with a demand matrix generated by a tree", Delft University of Technology, report20211012.

TUDelft

12

Inverse All Shortest Path Problem (IASPP)



$$A = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Shortest path (Dijkstra) ↓   ↑ ? Inverse all shortest paths

$$S = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 2 \\ 1 & 1 & 2 & 0 \end{bmatrix}$$

Given an $N \times N$ symmetric **demand matrix** D with zero diagonal elements, but positive off-diagonal elements

Determine a $N \times N$ **weighted adjacency matrix** $\tilde{A}$ such that

1. each shortest path weight obeys $s_{ij} \leq d_{ij}$
2. Variation 1: Optimized IASPP
   $||D - S||$ is minimized

   Descending Order Recovery (DOR)

   Variation 2: IASPP with link budget
   The sum of link weights $b = \sum_l w_l$ is fixed and the number L of links is minimized

   Omega-based Link Removal (OLR)

13

$$s_{ij} = w(\wp^*_{ij}) = \sum_{l \in \wp^*_{ij}} w_l$$

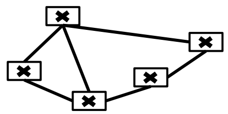
TU Delft

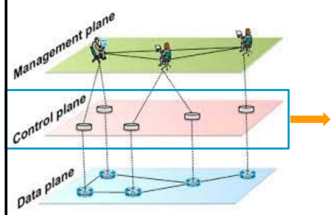
13

IASPP for Quality of Service in telecommunication

HUAWEI

The link weight w_{ij} from node i to node j denotes the maximum tolerable latency bound, e.g. bandwidth and delay, that a link l_{ij} should provide





1. Collect demand of End2End latency
2. Apply IASPP (DOR) to obtain an improved link weight structure
3. Communicate DOR improved links to devices in the network, e.g. via the NETCONF protocol

14

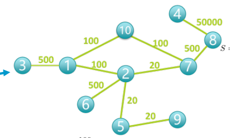
Schönwälder J, Björklund M, Shafer P. Network configuration management using NETCONF and YANG. IEEE communications magazine, 2010, 48(9): 166-173.

TU Delft

14

IASPP algorithms: DOR & OLR

DOR solves OIASPP

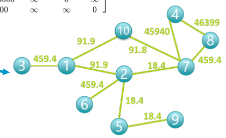


$$S = \begin{bmatrix} 0 & 100 & 500 & 50620 & 120 & \boxed{\mathbf{600}} & 120 & 620 & 140 & 100 \\ 100 & 0 & 600 & 50520 & 20 & 500 & 20 & 520 & 40 & 120 \\ 500 & 600 & 0 & 51120 & 620 & 1100 & 620 & 1120 & 640 & 600 \\ 50620 & 50520 & 51120 & 0 & 50540 & 51020 & 50500 & 50000 & 50560 & 50600 \\ 120 & 20 & 620 & 50540 & 0 & \boxed{\mathbf{520}} & 40 & 540 & 20 & 140 \\ \boxed{\mathbf{600}} & 500 & 1100 & 51020 & \boxed{\mathbf{520}} & 0 & 520 & 1020 & 540 & 620 \\ 120 & 20 & 620 & 50500 & 40 & 520 & 0 & 500 & \boxed{\mathbf{60}} & 100 \\ 620 & 520 & 1120 & 50000 & 540 & 1020 & 500 & 0 & 560 & 600 \\ 140 & 40 & 640 & 50560 & 20 & 540 & \boxed{\mathbf{60}} & 560 & 0 & 160 \\ 100 & 120 & 600 & 50600 & 140 & 620 & 100 & 600 & 160 & 0 \end{bmatrix}$$

$s_{ij} = d_{ij}$ except for those not achievable d_{ij} (e.g. violate the triangle rule) encircled in blue

$$D = \begin{bmatrix} 0 & 100 & 500 & \infty & \infty & 5000 & \infty & \infty & \infty & 100 \\ 100 & 0 & \infty & \infty & 20 & 500 & 20 & \infty & \infty & \infty \\ 500 & \infty & 0 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & 0 & \infty & \infty & \infty & 50000 & \infty & \infty \\ \infty & 20 & \infty & \infty & 0 & 1000 & \infty & \infty & 20 & \infty \\ 5000 & 500 & \infty & \infty & 1000 & 0 & \infty & \infty & \infty & \infty \\ \infty & 20 & \infty & \infty & \infty & \infty & 0 & 500 & 100000 & 100 \\ \infty & \infty & \infty & 50000 & \infty & \infty & 500 & 0 & \infty & \infty \\ \infty & \infty & \infty & \infty & 20 & \infty & 100000 & \infty & 0 & \infty \\ 100 & \infty & \infty & \infty & \infty & \infty & 100 & \infty & \infty & 0 \end{bmatrix}$$

OLR solves IASPP with link budget $\boldsymbol{b = 94048}$



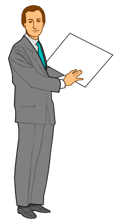
$$S = \begin{bmatrix} 0 & 91.88 & 459.4 & 46509 & 110.26 & 551.28 & 110.26 & 569.65 & 128.63 & 91.88 \\ 91.88 & 0 & 551.28 & 46418 & 18.376 & 459.4 & 18.376 & 477.77 & 36.752 & 110.26 \\ 459.4 & 551.28 & 0 & 46969 & 569.65 & 1010.7 & 569.65 & 1029.1 & 588.03 & 551.28 \\ 46509 & 46418 & 46969 & 0 & 46436 & 46877 & 46399 & 45940 & 46454 & 46491 \\ 110.26 & 18.376 & 569.65 & 46436 & 0 & 477.77 & 36.752 & 496.15 & 18.376 & 128.63 \\ 551.28 & 459.4 & 1010.7 & 46877 & 477.77 & 0 & 477.77 & 937.17 & 496.15 & 569.65 \\ 110.26 & 18.376 & 569.65 & 46399 & 36.752 & 477.77 & 0 & 459.4 & 55.128 & 91.88 \\ 569.65 & 477.77 & 1029.1 & 45940 & 496.15 & 937.17 & 459.4 & 0 & 514.53 & 551.28 \\ 128.63 & 36.752 & 588.03 & 46454 & 18.376 & 496.15 & 55.128 & 514.53 & 0 & 147.01 \\ 91.88 & 110.26 & 551.28 & 46491 & 128.63 & 569.65 & 91.88 & 551.28 & 147.01 & 0 \end{bmatrix}$$

$s_{ij} \leq d_{ij}$ and $\sum_l w_l = 94048 = b$

Zhihao Qiu, Ivan Jokic´, Siyu Tang, Rogier Noldus and Piet Van Mieghem
"The inverse all shortest path problem", submitted to IEEE TNSE

TU Delft

Outline



Inverse **linear** process:

Inverse all shortest path problem

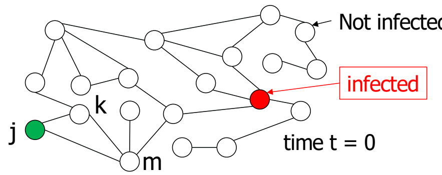
Inverse **non-linear** process:

Prediction of LRGE dynamics

TU Delft

SIS Virus spread in networks

Given:



Infection process: Poisson with infection strength β_{jk}
Curing process: Poisson with curing strength δ_j

Compute: Probability that node j is infected at time $t > 0$

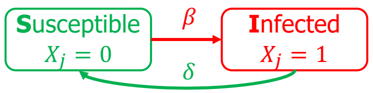
Assumptions:

1. SIS model: only 2 compartments: S & I
2. graph is static (not time-varying) and known
3. all processen are independent Poisson processes
4. infection and curing have constant strength (not time-varying, no mutations)

TUDelft

17

Markovian SIS epidemics in networks

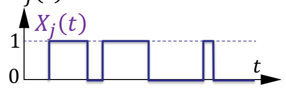


Markov state $X_j \in \{0,1\}$ of node j is a **Bernoulli random variable**

$$\Pr[X_j(t) = 1] = E[X_j(t)]$$

Each node j possesses a health state $X_j(t)$ at time t:

$X_j(t) = 0$: node j is not-infected at time t
$X_j(t) = 1$: node j is infected at time t



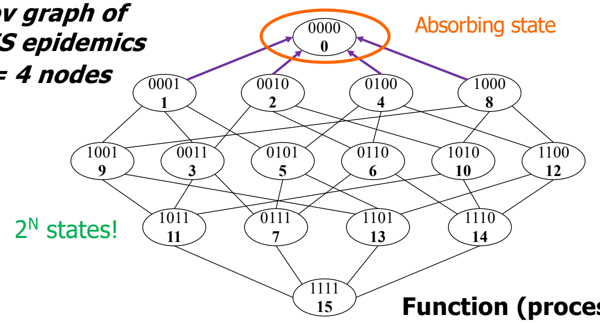
Infection probability of node j at time t: $v_j(t) = \Pr[X_j(t) = 1]$

18

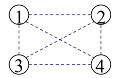
Infection process: Poisson with infection strength $\beta_{jk} = \beta$ (per link)
Curing process: Poisson with curing strength $\delta_j = \delta$ (per node)

18

Markov graph of the SIS epidemics on N = 4 nodes



Function (process)



Structure (graph)

P. Van Mieghem, J. Omic, R. E. Kooij, "Virus Spread in Networks", IEEE/ACM Transaction on Networking, Vol. 17, No. 1, pp. 1-14, (2009).

TU Delft

19

Governing Markovian SIS equation for node *j*

$$\frac{dE[X_j(t)]}{dt} = E\left[-\delta X_j(t) + (1 - X_j(t))\left\{\beta \sum_{k \in neighbor(j)} X_k(t)\right\}\right]$$

time-change of $E[X_j]=\Pr[X_j = 1]$ probability that node *j* is infected

if *infected* (X_j =1): probability of curing per unit time

if *not infected* (X_j =0): probability of infection per unit time from infected neighbors

$$\frac{dE[X_j]}{dt} = -\delta E[X_j] + \beta \sum_{k=1}^{N} a_{kj} E[X_k] - \beta \sum_{k=1}^{N} a_{kj} \boxed{E[X_j X_k]}$$

Complication

$E[X_j X_k]=\Pr[X_j = 1, X_k = 1]$

20

R. Pastor-Satorras, C. Castellano, P. Van Mieghem and A. Vespignani, "Epidemic processes in complex networks", Review of Modern Physics, Vol. 87, No. 3, pp. 925-979, 2015

TU Delft

20

Mean-field approximation: replace random variable by its mean

NIMFA

$$\frac{dE[X_j]}{dt} = E\left[-\delta X_j + (1 - X_j)\beta \sum_{k=1}^{N} a_{kj} X_k\right]$$

$$X_j \Longrightarrow E[\breve{X}_j] = w_j$$

$$\frac{dw_j}{dt} = -\delta w_j + (1 - w_j)\beta \sum_{k=1}^{N} a_{kj} w_k$$

From 2^N linear Markov differential equations to
N non-linear mean-field **approximating** diff. equations

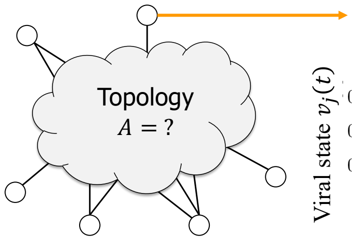
21

P. Van Mieghem, "The N-Intertwined SIS epidemic network model", Computing (Springer), Vol. 93, Issue 2, p. 147-169, 2011

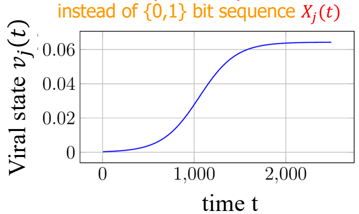
TU Delft

21

Graph Reconstruction from epidemics



Viral state observation = infection probability $v_j(t)$= $\Pr[X_j(t) = 1]$ over time, instead of $\{0,1\}$ bit sequence $X_j(t)$



Aim: Determine the N×N adjacency matrix A of the contact graph from a series of infection probabilities over time of all nodes

Solution: only partially possible

Prasse, B. and P. Van Mieghem, 2018, "Exact Network Reconstruction from Complete SIS Nodal State Infection Information Seems Infeasible", IEEE Transactions on Network Science and Engineering, Vol. 6, No. 4, October-December, pp. 748-759.

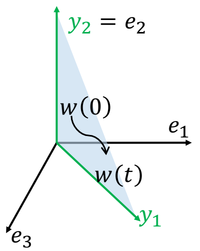
TU Delft

22

Agitation modes

Proper orthogonal decomposition (POD) of the viral state vector

$w(t) \approx \sum_{l=1}^{m} c_l(t) y_l$ $y_1, \dots, y_m$: orthonormal agitation modes
$c_l(t) = y_l^T w(t)$: scalar, projection of $w(t)$ on y_l



If the POD is accurate, we do not need N differential equations:

N differential equations $\frac{dw_i(t)}{dt} = f_{\text{SIS},i}(w(t)),$
$i = 1, \dots, N$

Projection on agitation modes y_l

$m < N$ differential equations $\frac{dc_l(t)}{dt} \approx y_l^T f_{\text{SIS}}(\sum_{l=1}^{m} c_l(t) y_l), \quad l = 1, \dots, m$

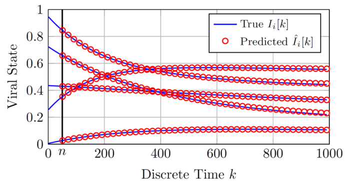
Prasse, B. and P. Van Mieghem, 2022, "Predicting network dynamics without requiring the knowledge of the interaction graph", Proceedings of the National Academy of Sciences (PNAS), Vol. 119, No. 44, e2205517119

TUDelft

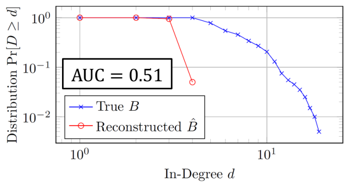
23

Accurate prediction of epidemic outbreaks *without* accurate network reconstruction

Prediction



Reconstruction



Process: only a few agitation modes

Graph: nearly all eigenmodes

Basis of the Network Inference Prediction Algorithm (**NIPA**)
Real-time data loading from RIVM (Dutch ministry of health):
https://www.nas.ewi.tudelft.nl/nipa/covid-prediction

B. Prasse and P. Van Mieghem, 2020, "Network Reconstruction and Prediction of Epidemic Outbreaks for General Group-Based Compartmental Epidemic Models", IEEE Transactions on Network Science and Engineering, Vol. 7, No. 4, October-December, pp. 2755-2764

UDelft

24

Summary: Inverse problems

- Inverse all shortest path problem (IASPP):
  - Broad application range
  - Exact solution for flow networks
    - effective resistance matrix for any graph are one-to-one coupled with the adjacency matrix (Fiedler's block inverse)
  - Path networks are generally challenging: two proposed solutions
    - DOR (descending order recovery)
    - Omega-based Link Removal (OLR)

- Prediction of "local-rule, global emergent" dynamics:
  - Possible without knowing the (assumed fixed) interaction graph!
  - Explanation of success of "deep learning methods"
    - Autonomous dynamic in high dimensions only evolves in a small subspace

25

TU Delft

25

Network Architectures and Services 26

NAS

TU Delft

26