INTRAS

Co-funded by the
Erasmus+ Programme
of the European Union

“Mathematical Methods in Transportation”

Syllabus

Prof. Kyandoghere Kyamakya

Prof. Jean Chamberlain Chedjou

UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master’s Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

CONTENTS

- Importance of Mathematics in Transportation
- Selected applications of mathematics in transportation
- Main objective of the Course: Knowledge & usability
- Important knowledge to be provided per chapter
- Spin-off of the course: Projects & Scientific contributions
- Didactic example: “Lecturing of a full chapter of this Course"

INTRAS

Co-funded by the
Erasmus+ Programme
of the European Union

Why is ”MATHEMATICS” so important in “TRANSPORTATION” ?

UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master’s Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

Mathematics as important instrument for improving traffic and transport

Improving traffic and transport using mathematics is an old dream (1900‘s), which is nowadays effective in many disciplines (or fields of Science) through the use of models, concepts, algorithms, and tools developed in the frame of **„OPERATIONS RESEARCH (From 1960s to date)”**

Applied mathematics:
Example: mathematical modeling of Traffic level of details

Continuous & Discrete Optimization
Example: Use of Neural networks to solve key problems in transportation

Statistics & stochasticity:
Example: Analysis of randomness and Forecasting algrithms in transportation

Traffic level of details
1) Microscopic (ODE)
2) Macroscopic (PDE)
3) Mesoscopic & Nanoscopic

Neural Networks concepts for:
1) Schedulling (in SCN and RT)
2) Travel-time\Energy\Pollution
3) SPP \ SPST \ MST \ TSP\ Flow

Applications in Transportation
1) Prediction
2) Proactivity
3) Performances

Co-funded by the
Erasmus+ Programme
of the European Union

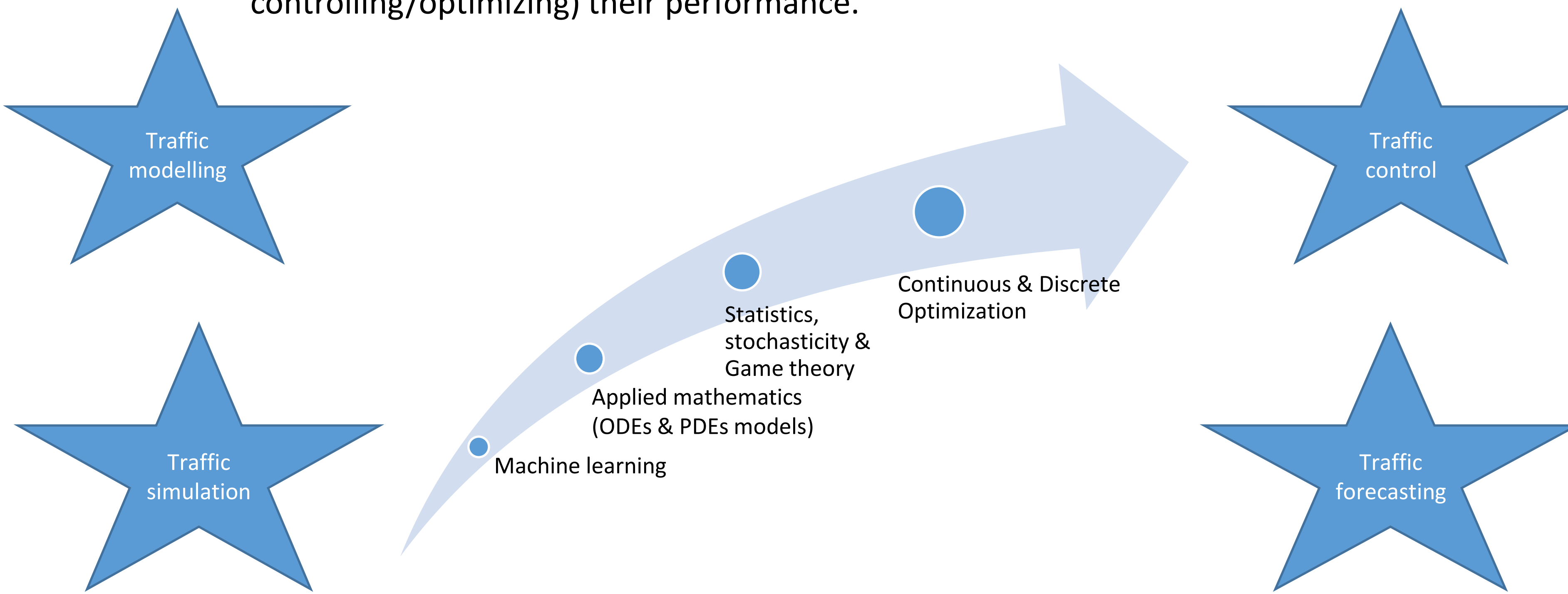
What is ”OPERATIONS RESEARCH” and why is it so important in the field of “TRANSPORTATION” ?

Intelligent Transport Systems: New ICT – based Master’s Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

Operations reseach in transportation

Operations research (O.R.) **:**

1. O.R. performs decision-making based on analytical methods (mathematical models), statistics and algorithms (simulation models).
2. O.R. is used to analyze complex real-world systems with the aim of improving (i.e., controlling/optimizing) their performance.



Co-funded by the
Erasmus+ Programme
of the European Union

What is the value/importance of ”this Lecture/Course” for the “New ICT- Master Program” ?

Sample Applications of Mathematics in transportation Engineering

❑ The knowledge acquired in this Course can be used to address the following issues:

Transportation planning

Example: Designs to facilitate the move of people & goods from s→t (•OD-Matrices, •Optimal path planning, • SPP, • MST, • Max Flow, etc.)

Design of Road-Networks

- Graph theory
- Lanes & Road capacity
- Traffic jam avoidance
- Travel time optimization

Design of Rail-Networks

- Graph theory
- Rails- Scheduling
- Collision avoidance through
- Mov. Synchro. on Rail Tracks

Traffic sensing (Sensor Data)

- Traffic counting
- Traffic tracking/visualization
- Traffic control

Traffic control strategies

- Pretimmed
- Actuated
- Semi-actuated

Traffic simulation Models/concepts

- Design of SimModels based- ODEs & PDEs
- Neural networks

Co-funded by the
Erasmus+ Programme
of the European Union

Important aspects to be covered in this Course/Lecture by each Lecturer

Important aspects to be investigated in each chapter

→ (1)

- Chapter 1. General introduction
  - ✓ Why is mathematics important in transportation?
  - ✓ How should mathematics be used in transportation?
  - ✓ Principle of modelling transportation systems mathematically
  - ✓ Presentation of concrete systems/scenarios in transportation along with their corresponding mathematical models
  - ✓ Examples are selected in Railway transportation, Road transportation, and in Supply Chain Networks (SCN) and Logistics

- Chapter 2. Basics of graph theory and applications in transportation
  - ✓ Selected applications of graph theory in transportation.
  - ✓ Basic concepts in graph theory
  - ✓ Description of SPP, SPST, MST, TSP, and Max Flow.
  - ✓ Dijkstra algorithm for SPST & MST in graph networks
  - ✓ Matrix- representation (e.g., adjacency-, Incidence-, Circuit- matrices) of graph networks with concrete examples in transportation.

Co-funded by the
Erasmus+ Programme
of the European Union

Important aspects to be investigated in each chapter

→ (2)

Chapter 3. Mathematical modeling of traffic flow

- ✓ Fundamental parameters of traffic flow: Speed-Flow-Density
- ✓ Mathematical modeling of the fundamental parameters of traffic flow
- ✓ Mathematical modeling of traffic flow on a multilane segment (with both overtaking and ramps).
- ✓ Mathematical modeling of wheel pairs movement of a rail vehicles
- ✓ Mathematical modeling of Railway Rescheduling Problem

Chapter 4. Basics of traffic signals control at isolated junction

- ✓ Performance criteria of a junction
- ✓ Mathematical model of a traffic junction
- ✓ Identification of a traffic junction
- ✓ Protected- and Unprotected- turns; Critical lane concept
- ✓ Graphs for traffic lanes and lane- groups
- ✓ Graphs for road intersections

Important aspects to be investigated in each chapter

→ (3)

- **Chapter 5. Mathematical modelling of scenarios/events in Railway transp.**
  - ✓ Graphical models of specific examples in Railway transportation
    - (see projects below)
  - ✓ Mathematical models of specific examples in Railway transportation
    - (see projects below)

- **Chapter 6. Basics of supply chain networks (SCN) and modelling principles**
  - ✓ Structure of a SCN & Framework for SCM
  - ✓ Design principle of a SCN
  - ✓ Graphical modeling of a SCN (see projects below)
  - ✓ Mathematical modelling of a SCN (see projects below)

Co-funded by the Erasmus+ Programme of the European Union

Important aspects to be investigated in each chapter

→ (4)

- **Chapter 7. MATLAB-CODING: Numerical simulation of Microscopic, Macroscopic and Mesoscopic traffic dynamics**
  - ✓ Microscopic traffic (ODEs)
  - ✓ Macroscopic traffic (PDEs)
  - ✓ Mesoscopic traffic (Coupled ODE & PDE system)

- **Chapter 8. LAB.-DEMO: SYNCHRO 7 & 9: Design of traffic junctions with different control strategies using SYNCHRO and measurement of the performance criteria**
  - ✓ Pretimed control
  - ✓ Actuated control
  - ✓ Semi-actuated control
  - ✓ Roundabout

Co-funded by the
Erasmus+ Programme
of the European Union

What is ”The Scientific value/importance of this Lecture/Course” ?

Spin-off of the course/Lecture

- The knowledge acquired in this Course can be used in the following key projects:

Fundamental Research Projects

Findings are: Development of

- ✓ Traffic Simulation models
- ✓ Traffic control strategies
- ✓ Traffic forecasting models

Scientific Publications

Expected contribution are:

- ✓ Conference papers
- ✓ Book chapters
- ✓ Journal papers
- ✓ Master Theses
- ✓ PhD Theses

Experimental Research Projects

Outcome is: Implementation of

- ✓ Traffic SimTools
- ✓ Traffic control devices
- ✓ Traffic sensors (e.g. devices for adaptive cruise control, tracking & detection, traffic counting, etc.)

INTRAS

Co-funded by the Erasmus+ Programme of the European Union

Selected concrete application examples of scenarios in transportation with related mathematical models

UNIVERSITÄT KLAGENFURT

INTRAS

Co-funded by the Erasmus+ Programme of the European Union

"A TYPICAL EXAMPLE OF APPLYING MATHEMATICS IN ROAD TRAFFIC"

How maths and driverless cars could spell the end of traffic jams

[Source]: Article by Lorna Wilson, Commercial Research Associate, University of Bath, **2016**

Macroscopic model

$$\frac{\partial Q}{\partial x} + \frac{\partial K}{\partial t} = 0$$

$Q = KU$

$Q(x,t)$

$K(x,t)$

$U(x,t)$

"A TYPICAL EXAMPLE OF APPLYING MATHEMATICS IN ROAD TRAFFIC"

Equation: Factors for Predicting Phantom Traffic Jams

[Source]: University of Alberta and MIT, 2011

β A measure of road conditions and driver behavior. The curvier the road or the nastier the weather, the longer it takes to slow down. That makes this number smaller and reduces the number of cars needed to make a jam pop up.

100 ß

ρ_L The lowest traffic density (in cars per unit of distance) at which a jam can occur. If at least this many cars are on the road, watch out.

$$\rho_L = \frac{\rho_M}{2}\left(1 - \sqrt{1 - \frac{4\beta}{u^2}}\right)$$

ρ_M The maximum traffic density for the particular road we're talking about, bumper-to-bumper.

20 cars per mile

– This minus sign configures the equation to give the lowest density at which a phantom jam can occur. But there's a highest density, too. Replace the minus with a plus and you get a maximum. If there are more than this many cars on the road, everyone is forced to go too slowly to trigger ghost gridlock.

u The speed you'd be going if you weren't in this damn traffic jam.

55 mph

1. This mathematical model was developed by a team of MIT mathematicians to describe the formation of "phantom jams"

2. This equation/ Model **(developed at the University of Alberta and MIT)** can tell you when to prepare for frustration (due to jam traffic)

A frustrated driver in a traffic jam

Co-funded by the Erasmus+ Programme of the European Union

“A TYPICAL EXAMPLE OF APPLYING MATHEMATICS IN ROAD TRAFFIC”

- This course analyses the following models: Greenshields, Daganzo, Smulders, and Drake:

Greenshields model 1

$$q = u_f k - \frac{u_f}{k_j} k^2$$

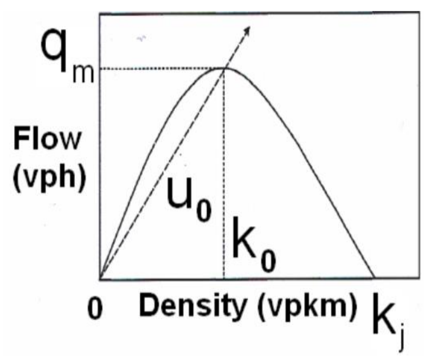
(Flow versus Density)

$$q_m = \frac{k_j u_f}{4}$$

(Maximum flow) (Capacity)

$$u_o = \frac{u_f}{2}$$

(Speed at capacity)



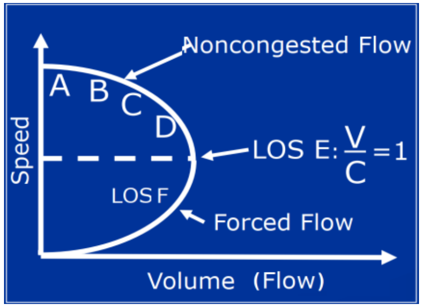
Greenshields model 2

$$u^2 = u_f u - \frac{u_f}{k_j} q$$

(Speed versus Flow)

$$\text{LOS E: } \frac{V}{C} = 1$$

(Level of Service (LOS) at capacity)



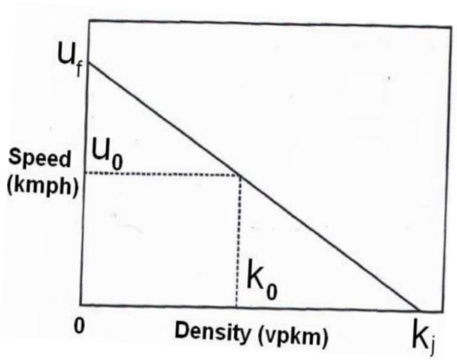
Greenshields model 3

$$u = u_f - \frac{u_f}{k_j} k$$

(Speed versus Density)

$$k_o = \frac{k_j}{2}$$

(Density at capacity)



INTRAS

Co-funded by the Erasmus+ Programme of the European Union

"A TYPICAL EXAMPLE OF APPLYING MATHEMATICS IN RAIL TRAFFIC"

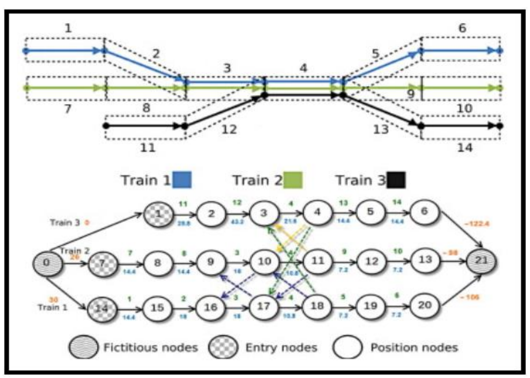
- This course analyses the following issue (amongst many others at stake in Rail traffic):

Real Time Rescheduling of Railway Networks

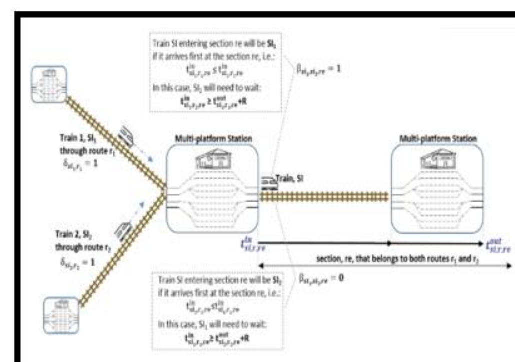
Discrete optimization

Mathematics in traffic and transport ...

Continuous optimization



Multi objective optimization



UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master's Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

"A TYPICAL EXAMPLE OF APPLYING MATHEMATICS IN SCN"

- This course also investigates the mathematical modeling of SCN

BIFURCATION ANALYSIS AND SYNCHRONIZATION ISSUES IN A THREE ECHELON SUPPLY CHAIN NETWORK

K. R. Anne, J. C. Chedjou & K. Kyamakya

International Journal of Logistics Research and Applications
A Leading Journal of Supply Chain Management Volume 12, 2009

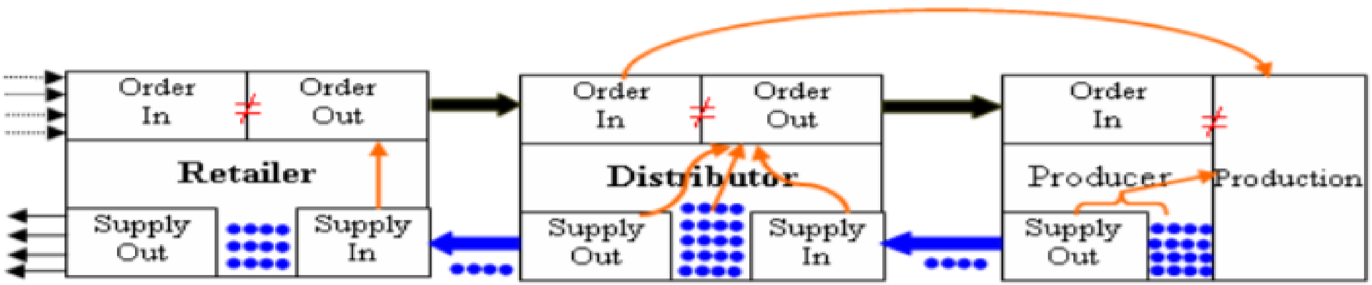


Figure 1: Three level supply chain model

- The quantity demanded by customers

$$x_i = m(y_{i-1} - x_{i-1})$$

- The inventory level of distributors

$$y_i = x_{i-1}(r - z_{i-1})$$

- The quantity produced by producers

$$z_i = x_{i-1}y_{i-1} + kz_{i-1}$$

$$\dot{x}' = (\sigma + d\sigma)(y' - x') + d_1$$

$$\dot{y}' = (r + dr)x' - y' - x'z' + d_2$$

$$\dot{z}' = x'y' - (b + db)z' + d_3$$

Co-funded by the
Erasmus+ Programme
of the European Union

Selected important projects to be considered in the frame of this lecture

Intelligent Transport Systems: New ICT – based Master's Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

Selected projects in railway transportation

- A Mathematical Model for Railway Control Systems - (NASA Contractor Report 198353)
- Mathematical approach applied to train scheduling
- Mathematical modeling of Railway Rescheduling Problem
- Mathematical model of wheel pairs movement of a rail vehicles
- Mathematical model for planning the distribution of locomotives to meet the demand for making up trains
- A mathematical model of the rail track presented as a bar on elastic and dissipative supports under the influence of moving loads

Rail traffic

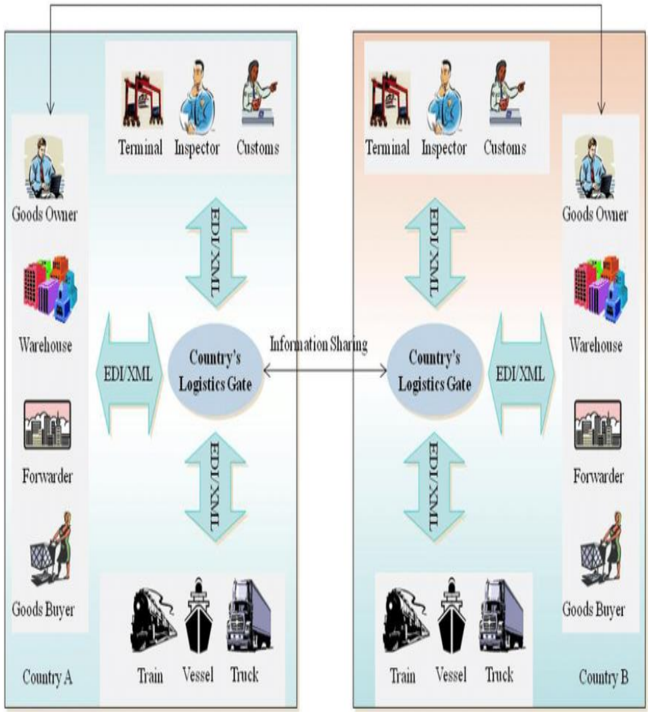
Selected projects in road transportation

- Mathematical modelling of traffic flow on highway
- Mathematical modeling of road transport in context of critical infrastructure protection
- Mathematical models for traffic control with concrete applications
- Mathematical modelling using integer linear programming approach for a Truck Rental Problem (TRP)
- A mathematical model in reduction of cost on transportation of sugarcane and the loss due to the accident in transportation
- A Neural Network Model for Drivers Lane-Changing Trajectory Prediction in Urban Traffic Flow

Road traffic

Selected projects in Supply chain networks

- Modelling Supply Chains with PDEs
- Modelling of a Supply Chain Network Driven By Stochastic Fluctuations
- Modelling of Job Shop Scheduling Problem
- Modelling and optimization of the assignment problem in a supply chain network
- Supply Chain Network Design under Uncertainty with Evidence Theory
- Formulating a mathematical model for container assignment optimization on an intermodal network



Supply chain networks & Logistics

INTRAS

Co-funded by the Erasmus+ Programme of the European Union

A didactic lecture on the topic:

"Mathematical modelling of Shortest path problems (SP) and Travelling salesman problem (TSP)

UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master's Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

INTRAS

Co-funded by the Erasmus+ Programme of the European Union

* Mathematical modeling of Shortest Path (SP) and Traveler Salesman Problem (TSP)

* Simulation algorithms based on the Neuro-dynamics concept (Neuro-computing)

Univ.-Prof. Dr.-Ing. Kyandoghere Kyamakya

Kyandoghere.Kyamakya@aau.at

Assoc. Prof. PD. Dr. Dr.-Ing. Jean Chamberlain Chedjou

jean.chedjou@aau.at

UNIVERSITÄT KLAGENFURT

Agreement number: 2017-3516/001-001

Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

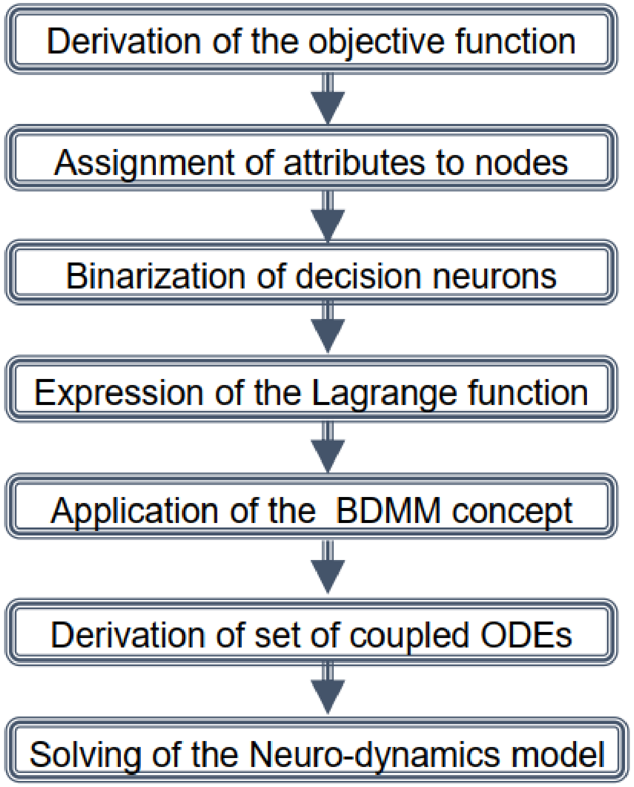
Literature

Journal Papers:

1. J. C. Platt and A. H. Barr, “Constrained differential optimization for neural networks,” *American Institute of Physics, Tech. Rep. TR-88-17, pp. 612-621, Apr. 1988.*
2. N. Alireza, O. Farahnaz, „An efficient dynamic model for solving the shortest path problem,“ *Transportation Research C, vol. 26, pp. 1–19, 2013.*
3. J. C. Chedjou and K. Kyamakya, „A Universal Concept for Robust Solving of Shortest Path Problems in Dynamically Reconfigurable Graphs,“ Hindawi Publishing Corporation, *Mathematical Problems in Engineering, Vol. 2015, Article ID345049, pp.23, 2015.*
4. J. C. Chedjou and K. Kyamakya, „Benchmarking a recurrent neural network based efficient shortest path problem (SPP) solver concept under difficult dynamic parameter settings conditions,“ *Neurocomputing (Elsevier), vol. 196, pp. 175–209, 2016*
5. J. C. Chedjou, K. Kyamakya and N. A. Akwir, “An Efficient, Scalable and Robust Neuro-Processor Based Concept for Solving Single-Cycle Traveling Salesman Problems in Complex and Dynamically Reconfigurable Graph Networks,” *IEEE Access, vol. 8, pp. 42297 - 42324, March 2020.*

Co-funded by the
Erasmus+ Programme
of the European Union

Transforming Graph Theoretical problems into set of ODEs: The Neurocomputing concept (BDMM)



Co-funded by the
Erasmus+ Programme
of the European Union

CASE STUDY 1:

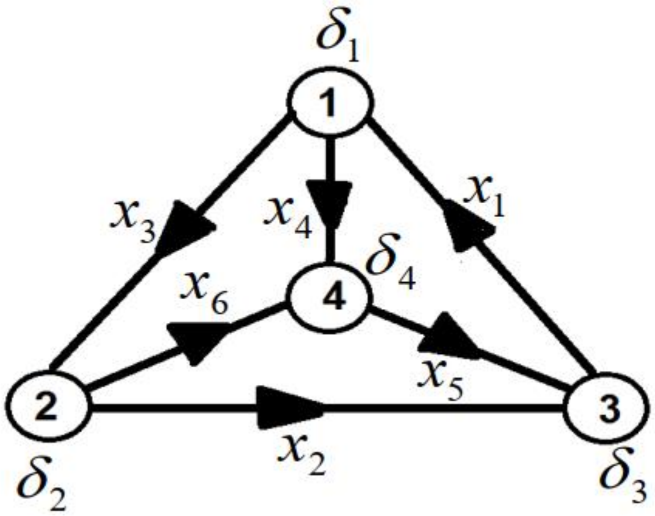
Shortest Path Problem (SPP)

„A directed graph of magnitude 4 and size 6“

Co-funded by the
Erasmus+ Programme
of the European Union

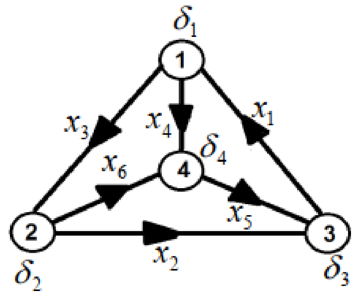
Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (1)

- Consider the directed graph of magnitude 4 and size 6 in Fig. 1:
  - ✓ Use the theory of optimization to model the SP problem in Fig. 1 mathematically.



Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (2)

- Mathematical modeling of the SP problem in Fig. 1.



Step 1: Expression of the total cost of the graph in Fig. 1 and objective function

$$Min\left[f(x_i, c_i) = (c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5 + c_6x_6)\right]$$

Step 2: Assignement of the three possible attributes to each node of Fig. 1

$$g(x_i, \delta_i) = \begin{cases} \text{Node1}: (x_3 + x_4 - x_1) = \delta_1 \\ \text{Node2}: (x_2 + x_6 - x_3) = \delta_2 \\ \text{Node3}: (x_1 - x_5 - x_2) = \delta_3 \\ \text{Node4}: (x_5 - x_6 - x_4) = \delta_4 \end{cases}$$

(Source Node: $\delta_i = 1$)

(Intermediate Node: $\delta_i = 0$)

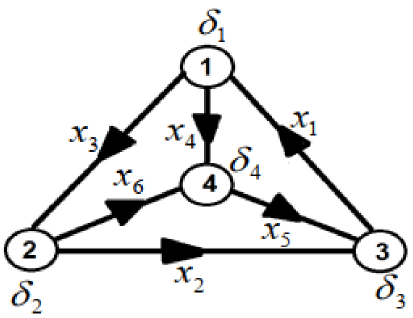
(Destination Node: $\delta_i = -1$)

Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (3)

- Mathematical modeling of the SP problem in Fig. 1.

Step 3: Expression of constraints related to binarization

$$h(x_i)=\begin{cases} x_1(x_1-1)=0 \\ x_2(x_2-1)=0 \\ x_3(x_3-1)=0 \\ x_4(x_4-1)=0 \\ x_5(x_5-1)=0 \\ x_6(x_6-1)=0 \end{cases}$$



Step 4: Expression of the Lagrange function as the total energy of the system

$$\begin{aligned} L &= (c_1x_1+c_2x_2+c_3x_3+c_4x_4+c_5x_5+c_6x_6)+\lambda_1(x_3+x_4-x_1-\delta_1) \\ &+\lambda_2(x_2+x_6-x_3-\delta_2)+\lambda_3(x_1-x_5-x_2-\delta_3)+\lambda_4(x_5-x_6-x_4-\delta_4) \\ &+\lambda_5x_1(x_1-1)+\lambda_6x_2(x_2-1)+\lambda_7x_3(x_3-1)+\lambda_8x_4(x_4-1)+\lambda_9x_5(x_5-1)+\lambda_{10}x_6(x_6-1) \end{aligned}$$

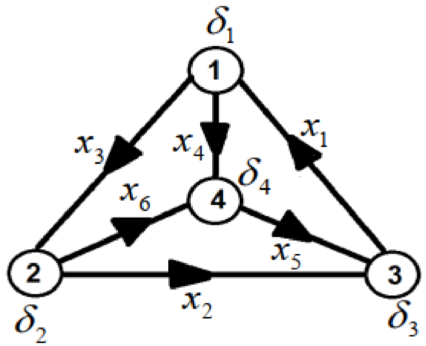
Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (4)

- Mathematical modeling of the SP problem in Fig. 1.

Step 5: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient descent to decision neurons

$$\begin{cases} \frac{dx_1}{dt} = -\frac{\partial L}{\partial x_1} = -[c_1 - \lambda_1 + \lambda_3 + \lambda_5(2x_1 - 1)] \\ \frac{dx_2}{dt} = -\frac{\partial L}{\partial x_2} = -[c_2 + \lambda_2 - \lambda_3 + \lambda_6(2x_2 - 1)] \\ \frac{dx_3}{dt} = -\frac{\partial L}{\partial x_3} = -[c_3 + \lambda_1 - \lambda_2 + \lambda_7(2x_3 - 1)] \\ \frac{dx_4}{dt} = -\frac{\partial L}{\partial x_4} = -[c_4 + \lambda_1 - \lambda_4 + \lambda_8(2x_4 - 1)] \\ \frac{dx_5}{dt} = -\frac{\partial L}{\partial x_5} = -[c_5 - \lambda_3 + \lambda_4 + \lambda_9(2x_5 - 1)] \\ \frac{dx_6}{dt} = -\frac{\partial L}{\partial x_6} = -[c_6 + \lambda_2 - \lambda_4 + \lambda_{10}(2x_6 - 1)] \end{cases}$$



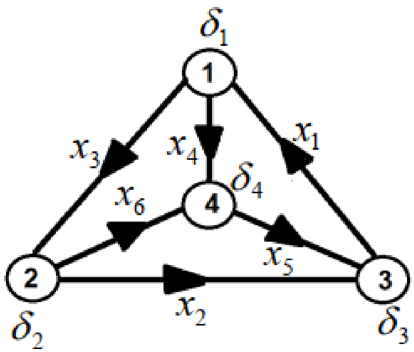
Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (5)

- Mathematical modeling of the SP problem in Fig. 1.

Step 6: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 1)

$$\begin{cases} \dfrac{d\lambda_1}{dt} = +\dfrac{\partial L}{\partial \lambda_1} = +\left[x_3 + x_4 - x_1 - \delta_1\right] \\ \dfrac{d\lambda_2}{dt} = +\dfrac{\partial L}{\partial \lambda_2} = +\left[x_2 + x_6 - x_3 - \delta_2\right] \\ \dfrac{d\lambda_3}{dt} = +\dfrac{\partial L}{\partial \lambda_3} = +\left[x_1 - x_5 - x_2 - \delta_3\right] \\ \dfrac{d\lambda_4}{dt} = +\dfrac{\partial L}{\partial \lambda_4} = +\left[x_5 - x_6 - x_4 - \delta_4\right] \end{cases}$$



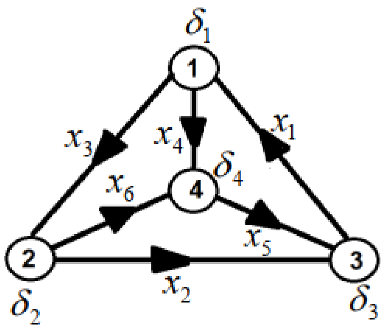
Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (6)

- Mathematical modeling of the SP problem in Fig. 1.

Step 7: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 2)

$$
\begin{cases}
\frac{d\lambda_5}{dt} = +\frac{\partial L}{\partial \lambda_5} = +[x_1(x_1-1)] \\
\frac{d\lambda_6}{dt} = +\frac{\partial L}{\partial \lambda_6} = +[x_2(x_2-1)] \\
\frac{d\lambda_7}{dt} = +\frac{\partial L}{\partial \lambda_7} = +[x_3(x_3-1)] \\
\frac{d\lambda_8}{dt} = +\frac{\partial L}{\partial \lambda_8} = +[x_4(x_4-1)] \\
\frac{d\lambda_9}{dt} = +\frac{\partial L}{\partial \lambda_9} = +[x_5(x_5-1)] \\
\frac{d\lambda_{10}}{dt} = +\frac{\partial L}{\partial \lambda_{10}} = +[x_6(x_6-1)]
\end{cases}
$$



Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (7)

- Mathematical modeling of the SP problem in Fig. 1.

Step 8: Neurocomputing concept- Derivation of the set of coupled ODEs model

Group 1 : Decision Neurons

$$\begin{cases} \frac{dx_1}{dt} = -[c_1 - \lambda_1 + \lambda_3 + \lambda_5(2x_1 - 1)] \\ \frac{dx_2}{dt} = -[c_2 + \lambda_2 - \lambda_3 + \lambda_6(2x_2 - 1)] \\ \frac{dx_3}{dt} = -[c_3 + \lambda_1 - \lambda_2 + \lambda_7(2x_3 - 1)] \\ \frac{dx_4}{dt} = -[c_4 + \lambda_1 - \lambda_4 + \lambda_8(2x_4 - 1)] \\ \frac{dx_5}{dt} = -[c_5 - \lambda_3 + \lambda_4 + \lambda_9(2x_5 - 1)] \\ \frac{dx_6}{dt} = -[c_6 + \lambda_2 - \lambda_4 + \lambda_{10}(2x_6 - 1)] \end{cases}$$

Group 2 : Multiplier Neurons

$$\begin{cases} \frac{d\lambda_1}{dt} = +[x_3 + x_4 - x_1 - \delta_1] \\ \frac{d\lambda_2}{dt} = +[x_2 + x_6 - x_3 - \delta_2] \\ \frac{d\lambda_3}{dt} = +[x_1 - x_5 - x_2 - \delta_3] \\ \frac{d\lambda_4}{dt} = +[x_5 - x_6 - x_4 - \delta_4] \end{cases}$$

Group 3 : Multiplier Neurons

$$\begin{cases} \frac{d\lambda_5}{dt} = +[x_1(x_1 - 1)] \\ \frac{d\lambda_6}{dt} = +[x_2(x_2 - 1)] \\ \frac{d\lambda_7}{dt} = +[x_3(x_3 - 1)] \\ \frac{d\lambda_8}{dt} = +[x_4(x_4 - 1)] \\ \frac{d\lambda_9}{dt} = +[x_5(x_5 - 1)] \\ \frac{d\lambda_{10}}{dt} = +[x_6(x_6 - 1)] \end{cases}$$

Mathematical modeling of the shortest path problem in a directed graph of magnitude 4 and size 6 (8)

- Mathematical modeling of the SP problem in Fig. 1.

Step 8: Neurocomputing concept- Derivation of the set of coupled ODEs model

$$\lambda_1 = x_7\ ;\ \lambda_6 = x_{12}\ ;$$
$$\lambda_2 = x_8\ ;\ \lambda_7 = x_{13}\ ;$$
$$\lambda_3 = x_9\ ;\ \lambda_8 = x_{14}\ ;$$
$$\lambda_4 = x_{10}\ ;\ \lambda_9 = x_{15}\ ;$$
$$\lambda_5 = x_{11}\ ;\ \lambda_{10} = x_{16}\ ;$$

$\text{Group 1: Decision Neurons}$

$$\begin{cases} \dfrac{dx_1}{dt} = -[c_1 - x_7 + x_9 + x_{11}(2x_1 - 1)] \\ \dfrac{dx_2}{dt} = -[c_2 + x_8 - x_9 + x_{12}(2x_2 - 1)] \\ \dfrac{dx_3}{dt} = -[c_3 + x_7 - x_8 + x_{13}(2x_3 - 1)] \\ \dfrac{dx_4}{dt} = -[c_4 + x_7 - x_{10} + x_{14}(2x_4 - 1)] \\ \dfrac{dx_5}{dt} = -[c_5 - x_9 + x_{10} + x_{15}(2x_5 - 1)] \\ \dfrac{dx_6}{dt} = -[c_6 + x_8 - x_{10} + x_{16}(2x_6 - 1)] \end{cases}$$

$\text{Group 2: Multiplier Neurons}$

$$\begin{cases} \dfrac{dx_7}{dt} = +[x_3 + x_4 - x_1 - \delta_1] \\ \dfrac{dx_8}{dt} = +[x_2 + x_6 - x_3 - \delta_2] \\ \dfrac{dx_9}{dt} = +[x_1 - x_5 - x_2 - \delta_3] \\ \dfrac{dx_{10}}{dt} = +[x_5 - x_6 - x_4 - \delta_4] \end{cases}$$

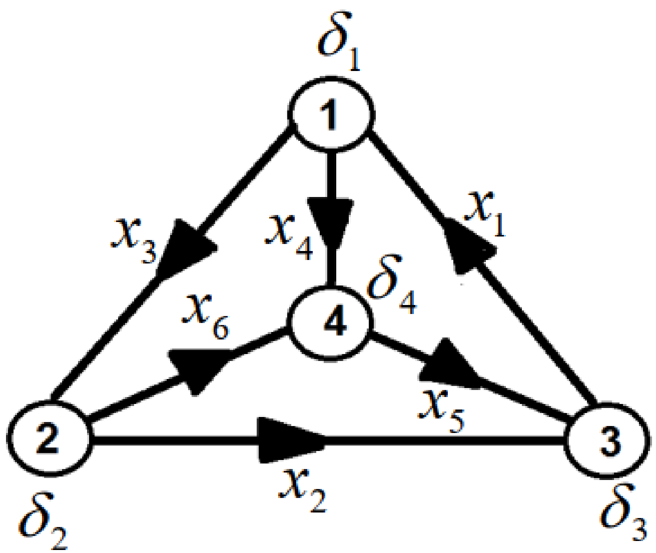
$\text{Group 3: Multiplier Neurons}$

$$\begin{cases} \dfrac{dx_{11}}{dt} = +[x_1(x_1 - 1)] \\ \dfrac{dx_{12}}{dt} = +[x_2(x_2 - 1)] \\ \dfrac{dx_{13}}{dt} = +[x_3(x_3 - 1)] \\ \dfrac{dx_{14}}{dt} = +[x_4(x_4 - 1)] \\ \dfrac{dx_{15}}{dt} = +[x_5(x_5 - 1)] \\ \dfrac{dx_{16}}{dt} = +[x_6(x_6 - 1)] \end{cases}$$

Intelligent Transport Systems: New ICT – based Master’s Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

MATLAB-CODING of the Neurocomputing concept

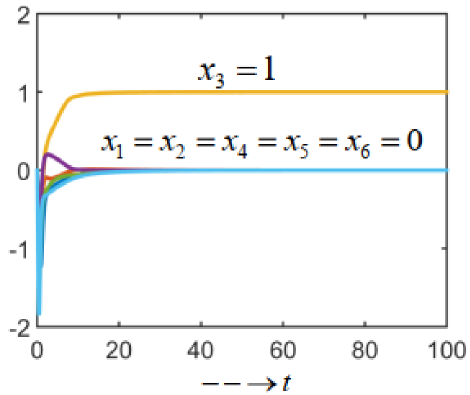
- Numerical simulation of the SP problem in Fig. 1.

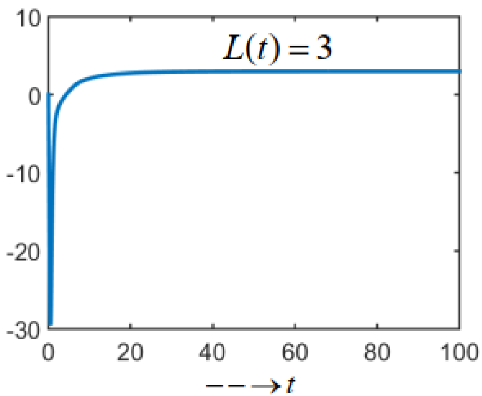



```
function dx=f(t,x)
dx=zeros(16,1);
c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=1; s2=0; s3=0; s4=-1;
dx(1)=-( c1 - x(7) + x(9) )-x(11)*(2*x(1)-1);
dx(2)=-( c2 + x(8) - x(9) )-x(12)*(2*x(2)-1);
dx(3)=-( c3 + x(7) - x(8) )-x(13)*(2*x(3)-1);
dx(4)=-( c4 + x(7) - x(10) )-x(14)*(2*x(4)-1);
dx(5)=-( c5 - x(9) + x(10) )-x(15)*(2*x(5)-1);
dx(6)=-( c6 + x(8) - x(10) )-x(16)*(2*x(6)-1);
dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(2) + x(6) - x(3) - s2 );
dx(9)=+( x(1) - x(5) - x(2) - s3 );
dx(10)=+( x(5) - x(6) - x(4) - s4 );
dx(11)=+x(1)*(x(1)-1);
dx(12)=+x(2)*(x(2)-1);
dx(13)=+x(3)*(x(3)-1);
dx(14)=+x(4)*(x(4)-1);
dx(15)=+x(5)*(x(5)-1);
dx(16)=+x(6)*(x(6)-1);
end
```

MATLAB-CODING of the Neurocomputing concept

- N1 (Source) and N2 (Destination)

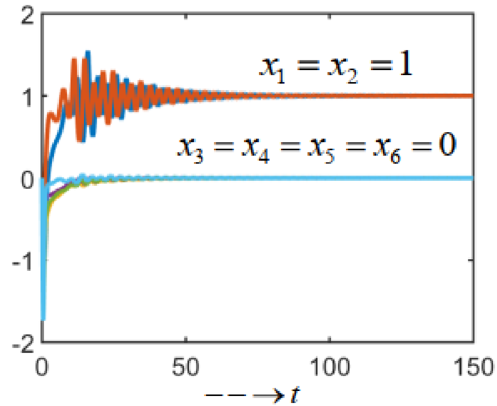


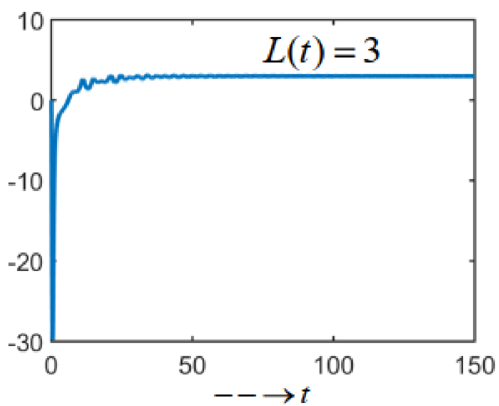



```
function dx=f(t,x)

dx=zeros(16,1);

c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=1; s2=-1; s3=0; s4=0;

dx(1)=-( c1 - x(7)  + x(9)  )-x(11)*(2*x(1)-1);
dx(2)=-( c2 + x(8)  - x(9)  )-x(12)*(2*x(2)-1);
dx(3)=-( c3 + x(7)  - x(8)  )-x(13)*(2*x(3)-1);
dx(4)=-( c4 + x(7)  - x(10) )-x(14)*(2*x(4)-1);
dx(5)=-( c5 - x(9)  + x(10) )-x(15)*(2*x(5)-1);
dx(6)=-( c6 + x(8)  - x(10) )-x(16)*(2*x(6)-1);

dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(2) + x(6) - x(3) - s2 );
dx(9)=+( x(1) - x(5) - x(2) - s3 );
dx(10)=+( x(5) - x(6) - x(4) - s4 );

dx(11)=+x(1)*(x(1)-1);
dx(12)=+x(2)*(x(2)-1);
dx(13)=+x(3)*(x(3)-1);
dx(14)=+x(4)*(x(4)-1);
dx(15)=+x(5)*(x(5)-1);
dx(16)=+x(6)*(x(6)-1);

end
```

MATLAB-CODING of the Neurocomputing concept

- N2 (Source) and N1 (Destination)

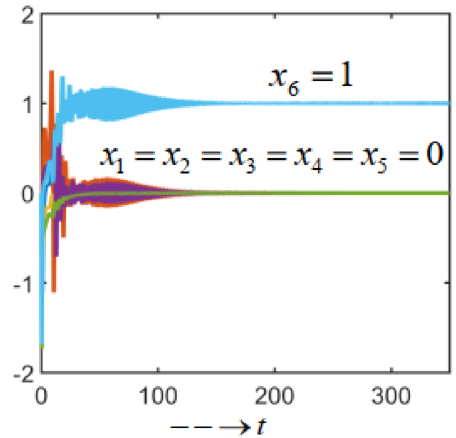


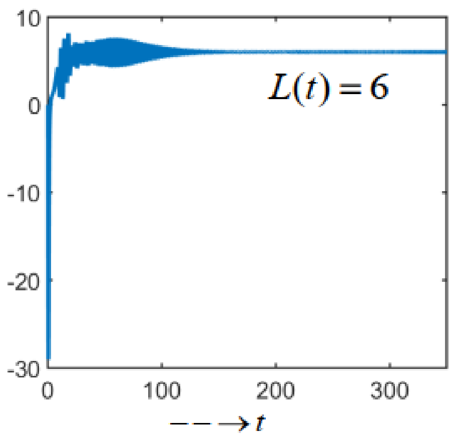



```
function dx=f(t,x)

dx=zeros(16,1);

c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=-1; s2=+1; s3=0; s4=0;

dx(1)=-( c1 - x(7) + x(9) )-x(11)*(2*x(1)-1);
dx(2)=-( c2 + x(8) - x(9) )-x(12)*(2*x(2)-1);
dx(3)=-( c3 + x(7) - x(8) )-x(13)*(2*x(3)-1);
dx(4)=-( c4 + x(7) - x(10) )-x(14)*(2*x(4)-1);
dx(5)=-( c5 - x(9) + x(10) )-x(15)*(2*x(5)-1);
dx(6)=-( c6 + x(8) - x(10) )-x(16)*(2*x(6)-1);

dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(2) + x(6) - x(3) - s2 );
dx(9)=+( x(1) - x(5) - x(2) - s3 );
dx(10)=+( x(5) - x(6) - x(4) - s4 );

dx(11)=+x(1)*(x(1)-1);
dx(12)=+x(2)*(x(2)-1);
dx(13)=+x(3)*(x(3)-1);
dx(14)=+x(4)*(x(4)-1);
dx(15)=+x(5)*(x(5)-1);
dx(16)=+x(6)*(x(6)-1);

end
```

MATLAB-CODING of the Neurocomputing concept

- N2 (Source) and N4 (Destination)

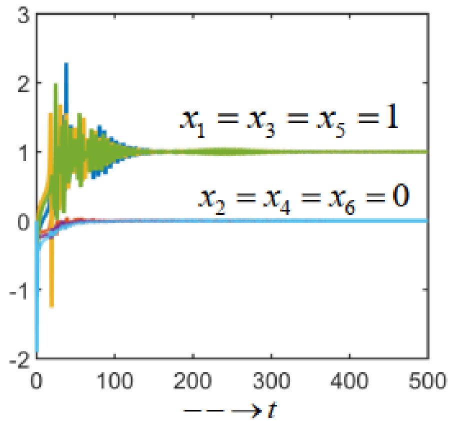


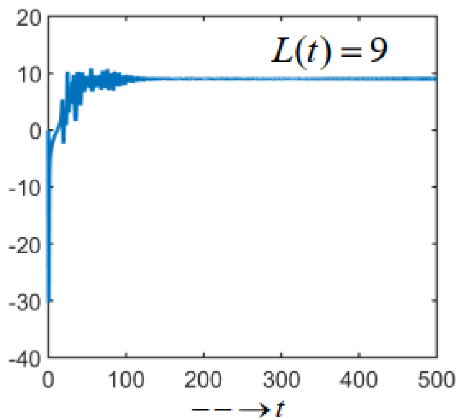



```
function dx=f(t,x)

dx=zeros(16,1);

c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=0; s2=1; s3=0; s4=-1;

dx(1)=-( c1 - x(7) + x(9) )-x(11)*(2*x(1)-1);
dx(2)=-( c2 + x(8) - x(9) )-x(12)*(2*x(2)-1);
dx(3)=-( c3 + x(7) - x(8) )-x(13)*(2*x(3)-1);
dx(4)=-( c4 + x(7) - x(10) )-x(14)*(2*x(4)-1);
dx(5)=-( c5 - x(9) + x(10) )-x(15)*(2*x(5)-1);
dx(6)=-( c6 + x(8) - x(10) )-x(16)*(2*x(6)-1);

dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(2) + x(6) - x(3) - s2 );
dx(9)=+( x(1) - x(5) - x(2) - s3 );
dx(10)=+( x(5) - x(6) - x(4) - s4 );

dx(11)=+x(1)*(x(1)-1);
dx(12)=+x(2)*(x(2)-1);
dx(13)=+x(3)*(x(3)-1);
dx(14)=+x(4)*(x(4)-1);
dx(15)=+x(5)*(x(5)-1);
dx(16)=+x(6)*(x(6)-1);

end
```

MATLAB-CODING of the Neurocomputing concept

- N4 (Source) and N2 (Destination)






```
function dx=f(t,x)

dx=zeros(16,1);

c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=0; s2=-1; s3=0; s4=1;

dx(1)=-( c1 - x(7) + x(9) )-x(11)*(2*x(1)-1);
dx(2)=-( c2 + x(8) - x(9) )-x(12)*(2*x(2)-1);
dx(3)=-( c3 + x(7) - x(8) )-x(13)*(2*x(3)-1);
dx(4)=-( c4 + x(7) - x(10) )-x(14)*(2*x(4)-1);
dx(5)=-( c5 - x(9) + x(10) )-x(15)*(2*x(5)-1);
dx(6)=-( c6 + x(8) - x(10) )-x(16)*(2*x(6)-1);

dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(2) + x(6) - x(3) - s2 );
dx(9)=+( x(1) - x(5) - x(2) - s3 );
dx(10)=+( x(5) - x(6) - x(4) - s4 );

dx(11)=+x(1)*(x(1)-1);
dx(12)=+x(2)*(x(2)-1);
dx(13)=+x(3)*(x(3)-1);
dx(14)=+x(4)*(x(4)-1);
dx(15)=+x(5)*(x(5)-1);
dx(16)=+x(6)*(x(6)-1);

end
```

INTRAS

Co-funded by the Erasmus+ Programme of the European Union

CASE STUDY 2:

Shortest Path Problem (SPP)

„Undirected graph of Magnitude 4 and size 12“

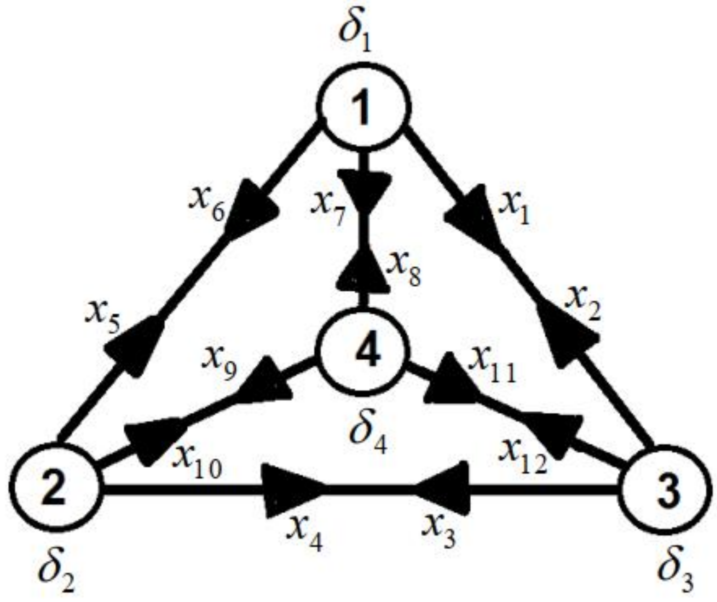
UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master’s Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

Co-funded by the Erasmus+ Programme of the European Union

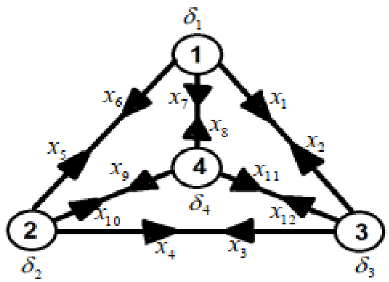
Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(1)

- Consider the undirected graph of magnitude 4 and size 12 in Fig. 2:
  - ✓ Use the theory of optimization to model the SP problem in Fig. 2 mathematically.



Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(2)

- Mathematical modeling of the SP problem in Fig. 2.



Step 1: Expression of the total cost of the graph in Fig. 2 and objective function

$$Min\Big[f(x_i, c_i) = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5 + c_6x_6 + c_7x_7 + c_8x_8 + c_9x_9 + c_{10}x_{10} + c_{11}x_{11} + c_{12}x_{12}\Big]$$

Step 2: Assignement of the three possible attributes to each node of Fig. 2

$$g(x_i, \delta_i) = \begin{cases} \text{Node1}: \ (x_1 + x_6 + x_7 - x_2 - x_5 - x_8) = \delta_1 \\ \text{Node2}: \ (x_4 + x_5 + x_{10} - x_3 - x_6 - x_9) = \delta_2 \\ \text{Node3}: \ (x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11}) = \delta_3 \\ \text{Node4}: \ (x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12}) = \delta_4 \end{cases}$$

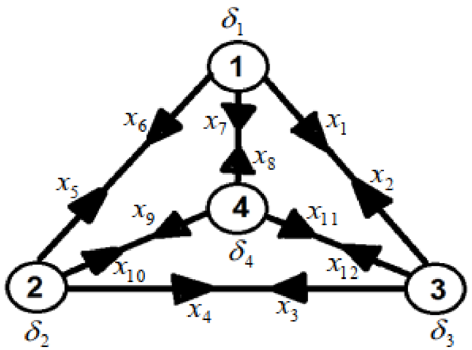
(Source Node: $\delta_i = 1$)
(Intermediate Node: $\delta_i = 0$)
(Destination Node: $\delta_i = -1$)

Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(3)

- Mathematical modeling of the SP problem in Fig. 2.

Step 3: Expression of constraints related to binarization

$$h(x_i)=\begin{cases} x_1(x_1-1)=0 \\ x_2(x_2-1)=0 \\ x_3(x_3-1)=0 \\ x_4(x_4-1)=0 \\ x_5(x_5-1)=0 \\ x_6(x_6-1)=0 \end{cases} \qquad h(x_i)=\begin{cases} x_7(x_7-1)=0 \\ x_8(x_8-1)=0 \\ x_9(x_9-1)=0 \\ x_{10}(x_{10}-1)=0 \\ x_{11}(x_{11}-1)=0 \\ x_{12}(x_{12}-1)=0 \end{cases}$$



Step 4: Expression of the Lagrange function as the total energy of the system

$$\begin{aligned} L = & \left(c_1x_1+c_2x_2+c_3x_3+c_4x_4+c_5x_5+c_6x_6+c_7x_7+c_8x_8+c_9x_9+c_{10}x_{10}+c_{11}x_{11}+c_{12}x_{12}\right) \\ & +\lambda_1\left(x_1+x_6+x_7-x_2-x_5-x_8-\delta_1\right)+\lambda_2\left(x_4+x_5+x_{10}-x_3-x_6-x_9-\delta_2\right) \\ & +\lambda_3\left(x_2+x_3+x_{12}-x_1-x_4-x_{11}-\delta_3\right)+\lambda_4\left(x_8+x_9+x_{11}-x_7-x_{10}-x_{12}-\delta_4\right) \\ & +\lambda_5x_1(x_1-1)+\lambda_6x_2(x_2-1)+\lambda_7x_3(x_3-1)+\lambda_8x_4(x_4-1)+\lambda_9x_5(x_5-1)+\lambda_{10}x_6(x_6-1) \\ & +\lambda_{11}x_7(x_7-1)+\lambda_{12}x_8(x_8-1)+\lambda_{13}x_9(x_9-1)+\lambda_{14}x_{10}(x_{10}-1)+\lambda_{15}x_{11}(x_{11}-1)+\lambda_{16}x_{12}(x_{12}-1) \end{aligned}$$

Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(4)

- Mathematical modeling of the SP problem in Fig. 2.

Step 5: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient descent to all twelve decision neurons

$$\begin{cases} \frac{dx_1}{dt} = -\frac{\partial L}{\partial x_1} = -[c_1 + \lambda_1 - \lambda_3 + \lambda_5(2x_1 - 1)] \\ \frac{dx_2}{dt} = -\frac{\partial L}{\partial x_2} = -[c_2 - \lambda_1 + \lambda_3 + \lambda_6(2x_2 - 1)] \\ \frac{dx_3}{dt} = -\frac{\partial L}{\partial x_3} = -[c_3 - \lambda_2 + \lambda_3 + \lambda_7(2x_3 - 1)] \\ \frac{dx_4}{dt} = -\frac{\partial L}{\partial x_4} = -[c_4 + \lambda_2 - \lambda_3 + \lambda_8(2x_4 - 1)] \\ \frac{dx_5}{dt} = -\frac{\partial L}{\partial x_5} = -[c_5 - \lambda_1 + \lambda_2 + \lambda_9(2x_5 - 1)] \\ \frac{dx_6}{dt} = -\frac{\partial L}{\partial x_6} = -[c_6 + \lambda_1 - \lambda_2 + \lambda_{10}(2x_6 - 1)] \end{cases}$$

$$\begin{cases} \frac{dx_7}{dt} = -\frac{\partial L}{\partial x_7} = -[c_7 + \lambda_1 - \lambda_4 + \lambda_{11}(2x_7 - 1)] \\ \frac{dx_8}{dt} = -\frac{\partial L}{\partial x_8} = -[c_8 - \lambda_1 + \lambda_4 + \lambda_{12}(2x_8 - 1)] \\ \frac{dx_9}{dt} = -\frac{\partial L}{\partial x_9} = -[c_9 - \lambda_2 + \lambda_4 + \lambda_{13}(2x_9 - 1)] \\ \frac{dx_{10}}{dt} = -\frac{\partial L}{\partial x_{10}} = -[c_{10} + \lambda_2 - \lambda_4 + \lambda_{14}(2x_{10} - 1)] \\ \frac{dx_{11}}{dt} = -\frac{\partial L}{\partial x_{11}} = -[c_{11} - \lambda_3 + \lambda_4 + \lambda_{15}(2x_{11} - 1)] \\ \frac{dx_{12}}{dt} = -\frac{\partial L}{\partial x_{12}} = -[c_{12} + \lambda_3 - \lambda_4 + \lambda_{16}(2x_{12} - 1)] \end{cases}$$

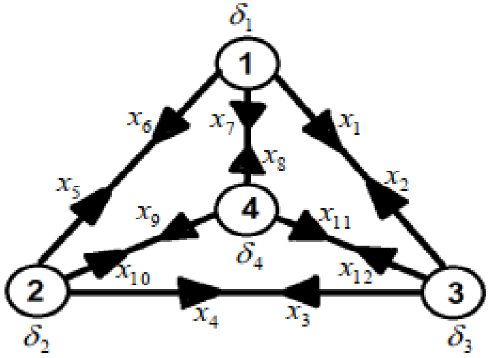
Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(5)

- Mathematical modeling of the SP problem in Fig. 2.

Step 6: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 1)

$$\begin{cases} \dfrac{d\lambda_1}{dt} = +\dfrac{\partial L}{\partial \lambda_1} = +\left[x_1 + x_6 + x_7 - x_2 - x_5 - x_8 - \delta_1\right] \\ \dfrac{d\lambda_2}{dt} = +\dfrac{\partial L}{\partial \lambda_2} = +\left[x_4 + x_5 + x_{10} - x_3 - x_6 - x_9 - \delta_2\right] \\ \dfrac{d\lambda_3}{dt} = +\dfrac{\partial L}{\partial \lambda_3} = +\left[x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11} - \delta_3\right] \\ \dfrac{d\lambda_4}{dt} = +\dfrac{\partial L}{\partial \lambda_4} = +\left[x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12} - \delta_4\right] \end{cases}$$



Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(6)

- Mathematical modeling of the SP problem in Fig. 2.

Step 7: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 2)

$$\begin{cases} \frac{d\lambda_5}{dt} = +\frac{\partial L}{\partial \lambda_5} = +[x_1(x_1-1)] \\ \frac{d\lambda_6}{dt} = +\frac{\partial L}{\partial \lambda_6} = +[x_2(x_2-1)] \\ \frac{d\lambda_7}{dt} = +\frac{\partial L}{\partial \lambda_7} = +[x_3(x_3-1)] \\ \frac{d\lambda_8}{dt} = +\frac{\partial L}{\partial \lambda_8} = +[x_4(x_4-1)] \\ \frac{d\lambda_9}{dt} = +\frac{\partial L}{\partial \lambda_9} = +[x_5(x_5-1)] \\ \frac{d\lambda_{10}}{dt} = +\frac{\partial L}{\partial \lambda_{10}} = +[x_6(x_6-1)] \end{cases} \qquad \begin{cases} \frac{d\lambda_{11}}{dt} = +\frac{\partial L}{\partial \lambda_{11}} = +[x_7(x_7-1)] \\ \frac{d\lambda_{12}}{dt} = +\frac{\partial L}{\partial \lambda_{12}} = +[x_8(x_8-1)] \\ \frac{d\lambda_{13}}{dt} = +\frac{\partial L}{\partial \lambda_{13}} = +[x_9(x_9-1)] \\ \frac{d\lambda_{14}}{dt} = +\frac{\partial L}{\partial \lambda_{14}} = +[x_{10}(x_{10}-1)] \\ \frac{d\lambda_{15}}{dt} = +\frac{\partial L}{\partial \lambda_{15}} = +[x_{11}(x_{11}-1)] \\ \frac{d\lambda_{16}}{dt} = +\frac{\partial L}{\partial \lambda_{16}} = +[x_{12}(x_{12}-1)] \end{cases}$$

Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(7)

- Mathematical modeling of the SP problem in Fig. 2.

Step 8: Neurocomputing concept- Derivation of the set of coupled ODEs model

Group 1 : Decision Neurons

$$
\begin{cases}
\dot{x}_1 = -[c_1 + \lambda_1 - \lambda_3 + \lambda_5(2x_1 - 1)] \\
\dot{x}_2 = -[c_2 - \lambda_1 + \lambda_3 + \lambda_6(2x_2 - 1)] \\
\dot{x}_3 = -[c_3 - \lambda_2 + \lambda_3 + \lambda_7(2x_3 - 1)] \\
\dot{x}_4 = -[c_4 + \lambda_2 - \lambda_3 + \lambda_8(2x_4 - 1)] \\
\dot{x}_5 = -[c_5 - \lambda_1 + \lambda_2 + \lambda_9(2x_5 - 1)] \\
\dot{x}_6 = -[c_6 + \lambda_1 - \lambda_2 + \lambda_{10}(2x_6 - 1)] \\
\dot{x}_7 = -[c_7 + \lambda_1 - \lambda_4 + \lambda_{11}(2x_7 - 1)] \\
\dot{x}_8 = -[c_8 - \lambda_1 + \lambda_4 + \lambda_{12}(2x_8 - 1)] \\
\dot{x}_9 = -[c_9 - \lambda_2 + \lambda_4 + \lambda_{13}(2x_9 - 1)] \\
\dot{x}_{10} = -[c_{10} + \lambda_2 - \lambda_4 + \lambda_{14}(2x_{10} - 1)] \\
\dot{x}_{11} = -[c_{11} - \lambda_3 + \lambda_4 + \lambda_{15}(2x_{11} - 1)] \\
\dot{x}_{12} = -[c_{12} + \lambda_3 - \lambda_4 + \lambda_{16}(2x_{12} - 1)]
\end{cases}
$$

Group 2 : Multiplier Neurons

$$
\begin{cases}
\dot{\lambda}_1 = +[x_1 + x_6 + x_7 - x_2 - x_5 - x_8 - \delta_1] \\
\dot{\lambda}_2 = +[x_4 + x_5 + x_{10} - x_3 - x_6 - x_9 - \delta_2] \\
\dot{\lambda}_3 = +[x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11} - \delta_3] \\
\dot{\lambda}_4 = +[x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12} - \delta_4]
\end{cases}
$$

Group 3 : Multiplier Neurons

$$
\begin{cases}
\dot{\lambda}_5 = +[x_1(x_1 - 1)] \\
\dot{\lambda}_6 = +[x_2(x_2 - 1)] \\
\dot{\lambda}_7 = +[x_3(x_3 - 1)] \\
\dot{\lambda}_8 = +[x_4(x_4 - 1)] \\
\dot{\lambda}_9 = +[x_5(x_5 - 1)] \\
\dot{\lambda}_{10} = +[x_6(x_6 - 1)] \\
\dot{\lambda}_{11} = +[x_7(x_7 - 1)] \\
\dot{\lambda}_{12} = +[x_8(x_8 - 1)] \\
\dot{\lambda}_{13} = +[x_9(x_9 - 1)] \\
\dot{\lambda}_{14} = +[x_{10}(x_{10} - 1)] \\
\dot{\lambda}_{15} = +[x_{11}(x_{11} - 1)] \\
\dot{\lambda}_{16} = +[x_{12}(x_{12} - 1)]
\end{cases}
$$

Mathematical modeling of the shortest path problem in an undirected graph of magnitude 4 and size 12(8)

- Mathematical modeling of the SP problem in Fig. 2.

Step 8: Neurocomputing concept- Derivation of the set of coupled ODEs model

Group 1 : Decision Neurons

$$\begin{cases}\dot{x}_1 = -[c_1 + x_{13} - x_{15} + x_{17}(2x_1 - 1)] \\ \dot{x}_2 = -[c_2 - x_{13} + x_{15} + x_{18}(2x_2 - 1)] \\ \dot{x}_3 = -[c_3 - x_{14} + x_{15} + x_{19}(2x_3 - 1)] \\ \dot{x}_4 = -[c_4 + x_{14} - x_{15} + x_{20}(2x_4 - 1)] \\ \dot{x}_5 = -[c_5 - x_{13} + x_{14} + x_{21}(2x_5 - 1)] \\ \dot{x}_6 = -[c_6 + x_{13} - x_{14} + x_{22}(2x_6 - 1)] \\ \dot{x}_7 = -[c_7 + x_{13} - x_{16} + x_{23}(2x_7 - 1)] \\ \dot{x}_8 = -[c_8 - x_{13} + x_{16} + x_{24}(2x_8 - 1)] \\ \dot{x}_9 = -[c_9 - x_{14} + x_{16} + x_{25}(2x_9 - 1)] \\ \dot{x}_{10} = -[c_{10} + x_{14} - x_{16} + x_{26}(2x_{10} - 1)] \\ \dot{x}_{11} = -[c_{11} - x_{15} + x_{16} + x_{27}(2x_{11} - 1)] \\ \dot{x}_{12} = -[c_{12} + x_{15} - x_{16} + x_{28}(2x_{12} - 1)]\end{cases}$$

$$\lambda_1 = x_{13}\,;\ \lambda_6 = x_{18}\,;\ \lambda_{11} = x_{23}\,;$$
$$\lambda_2 = x_{14}\,;\ \lambda_7 = x_{19}\,;\ \lambda_{12} = x_{24}\,;$$
$$\lambda_3 = x_{15}\,;\ \lambda_8 = x_{20}\,;\ \lambda_{13} = x_{25}\,;$$
$$\lambda_4 = x_{16}\,;\ \lambda_9 = x_{21}\,;\ \lambda_{14} = x_{26}\,;$$
$$\lambda_5 = x_{17}\,;\ \lambda_{10} = x_{22}\,;\ \lambda_{15} = x_{27}\,;$$
$$\lambda_{16} = x_{28}\,;$$

Group 2 : Multiplier Neurons

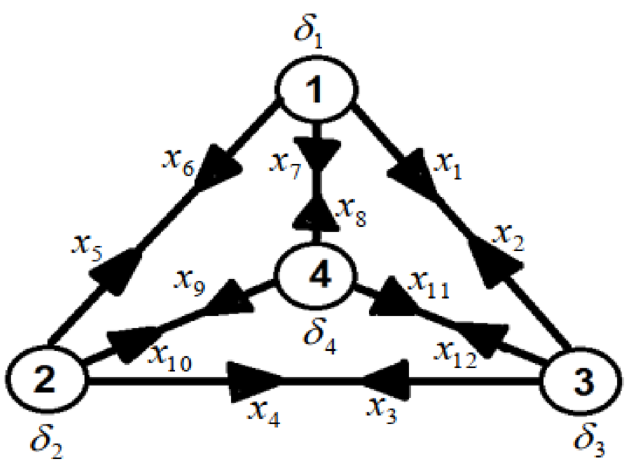
$$\begin{cases}\dot{x}_{13} = +[x_1 + x_6 + x_7 - x_2 - x_5 - x_8 - \delta_1] \\ \dot{x}_{14} = +[x_4 + x_5 + x_{10} - x_3 - x_6 - x_9 - \delta_2] \\ \dot{x}_{15} = +[x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11} - \delta_3] \\ \dot{x}_{16} = +[x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12} - \delta_4]\end{cases}$$

Group 3 : Multiplier Neurons

$$\begin{cases}\dot{x}_{17} = +[x_1(x_1 - 1)] \\ \dot{x}_{18} = +[x_2(x_2 - 1)] \\ \dot{x}_{19} = +[x_3(x_3 - 1)] \\ \dot{x}_{20} = +[x_4(x_4 - 1)] \\ \dot{x}_{21} = +[x_5(x_5 - 1)] \\ \dot{x}_{22} = +[x_6(x_6 - 1)] \\ \dot{x}_{23} = +[x_7(x_7 - 1)] \\ \dot{x}_{24} = +[x_8(x_8 - 1)] \\ \dot{x}_{25} = +[x_9(x_9 - 1)] \\ \dot{x}_{26} = +[x_{10}(x_{10} - 1)] \\ \dot{x}_{27} = +[x_{11}(x_{11} - 1)] \\ \dot{x}_{28} = +[x_{12}(x_{12} - 1)]\end{cases}$$

MATLAB-CODING of the Neurocomputing concept

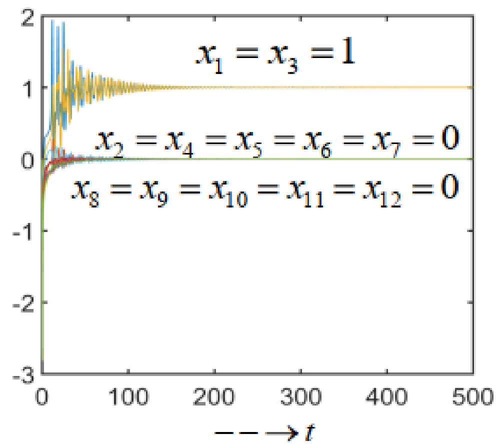
- Numerical simulation of the SP problem in Fig. 2.

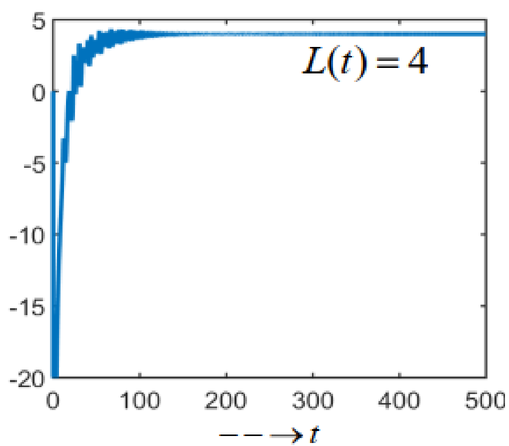



```
function dx=f(t,x)
dx=zeros(28,1);
c1=1;c2=2;c3=3;c4=4;c5=5;c6=6;c7=7;c8=8;c9=9;c10=10;
c11=11;c12=12;              s1=1; s2=-1; s3=0; s4=0;

dx(1)=-( c1 + x(13) - x(15) + x(17)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(15) + x(18)*(2*x(2)-1) );
dx(3)=-( c3 - x(14) + x(15) + x(19)*(2*x(3)-1) );
dx(4)=-( c4 + x(14) - x(15) + x(20)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(21)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) - x(14) + x(22)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) - x(16) + x(23)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(16) + x(24)*(2*x(8)-1) );
dx(9)=-( c9 - x(14) + x(16) + x(25)*(2*x(9)-1) );
dx(10)=-( c10 + x(14) - x(16) + x(26)*(2*x(10)-1) );
dx(11)=-( c11 - x(15) + x(16) + x(27)*(2*x(11)-1) );
dx(12)=-( c12 + x(15) - x(16) + x(28)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s2 );
dx(15)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s3 );
dx(16)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s4 );

dx(17)=+x(1)*(x(1)-1);
dx(18)=+x(2)*(x(2)-1);
dx(19)=+x(3)*(x(3)-1);
dx(20)=+x(4)*(x(4)-1);
dx(21)=+x(5)*(x(5)-1);
dx(22)=+x(6)*(x(6)-1);
dx(23)=+x(7)*(x(7)-1);
dx(24)=+x(8)*(x(8)-1);
dx(25)=+x(9)*(x(9)-1);
dx(26)=+x(10)*(x(10)-1);
dx(27)=+x(11)*(x(11)-1);
dx(28)=+x(12)*(x(12)-1);

end
```

MATLAB-CODING of the Neurocomputing concept

- N1 (Source) and N2 (Destination)

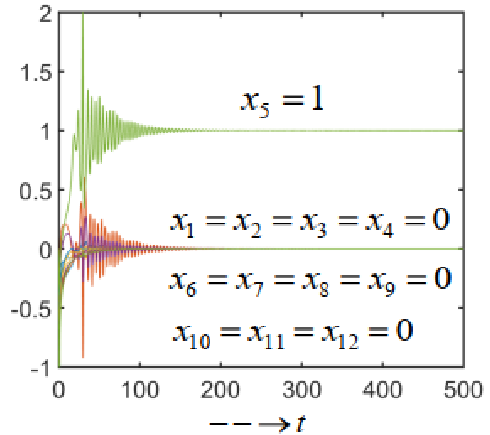


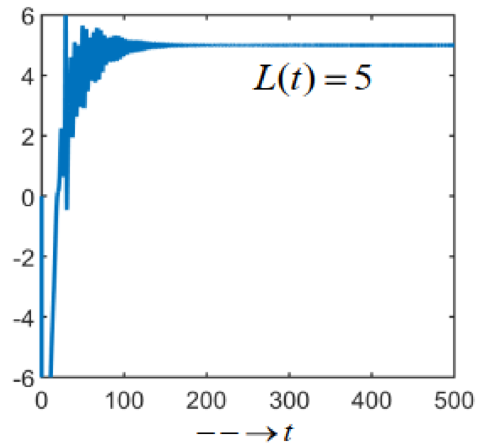



```
function dx=f(t,x)
dx=zeros(28,1);
c1=1;c2=2;c3=3;c4=4;c5=5;c6=6;c7=7;c8=8;c9=9;c10=10;
c11=11;c12=12;              s1=1; s2=-1; s3=0; s4=0;

dx(1)=-( c1 + x(13) - x(15) + x(17)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(15) + x(18)*(2*x(2)-1) );
dx(3)=-( c3 - x(14) + x(15) + x(19)*(2*x(3)-1) );
dx(4)=-( c4 + x(14) - x(15) + x(20)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(21)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) - x(14) + x(22)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) - x(16) + x(23)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(16) + x(24)*(2*x(8)-1) );
dx(9)=-( c9 - x(14) + x(16) + x(25)*(2*x(9)-1) );
dx(10)=-( c10 + x(14) - x(16) + x(26)*(2*x(10)-1) );
dx(11)=-( c11 - x(15) + x(16) + x(27)*(2*x(11)-1) );
dx(12)=-( c12 + x(15) - x(16) + x(28)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s2 );
dx(15)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s3 );
dx(16)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s4 );

dx(17)=+x(1)*(x(1)-1);
dx(18)=+x(2)*(x(2)-1);
dx(19)=+x(3)*(x(3)-1);
dx(20)=+x(4)*(x(4)-1);
dx(21)=+x(5)*(x(5)-1);
dx(22)=+x(6)*(x(6)-1);
dx(23)=+x(7)*(x(7)-1);
dx(24)=+x(8)*(x(8)-1);
dx(25)=+x(9)*(x(9)-1);
dx(26)=+x(10)*(x(10)-1);
dx(27)=+x(11)*(x(11)-1);
dx(28)=+x(12)*(x(12)-1);
end
```

MATLAB-CODING of the Neurocomputing concept

- N2 (Source) and N1 (Destination)



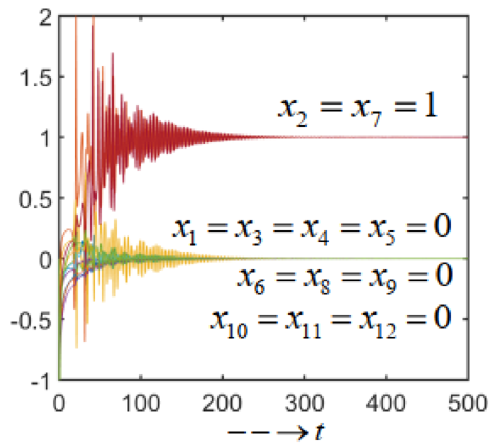


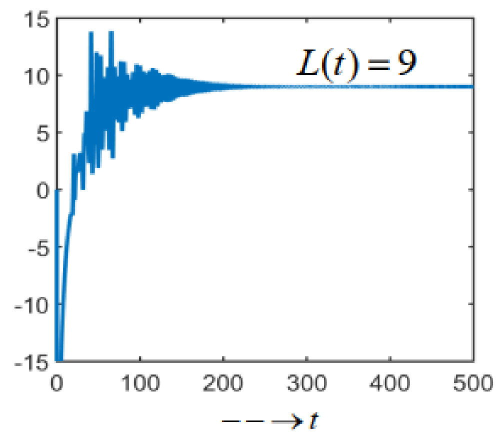

```
function dx=f(t,x)
dx=zeros(28,1);
c1=1;c2=2;c3=3;c4=4;c5=5;c6=6;c7=7;c8=8;c9=9;c10=10;
c11=11;c12=12;            s1=-1; s2=1; s3=0; s4=0;

dx(1)=-( c1 + x(13) - x(15) + x(17)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(15) + x(18)*(2*x(2)-1) );
dx(3)=-( c3 - x(14) + x(15) + x(19)*(2*x(3)-1) );
dx(4)=-( c4 + x(14) - x(15) + x(20)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(21)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) - x(14) + x(22)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) - x(16) + x(23)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(16) + x(24)*(2*x(8)-1) );
dx(9)=-( c9 - x(14) + x(16) + x(25)*(2*x(9)-1) );
dx(10)=-( c10 + x(14) - x(16) + x(26)*(2*x(10)-1) );
dx(11)=-( c11 - x(15) + x(16) + x(27)*(2*x(11)-1) );
dx(12)=-( c12 + x(15) - x(16) + x(28)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s2 );
dx(15)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s3 );
dx(16)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s4 );

dx(17)=+x(1)*(x(1)-1);
dx(18)=+x(2)*(x(2)-1);
dx(19)=+x(3)*(x(3)-1);
dx(20)=+x(4)*(x(4)-1);
dx(21)=+x(5)*(x(5)-1);
dx(22)=+x(6)*(x(6)-1);
dx(23)=+x(7)*(x(7)-1);
dx(24)=+x(8)*(x(8)-1);
dx(25)=+x(9)*(x(9)-1);
dx(26)=+x(10)*(x(10)-1);
dx(27)=+x(11)*(x(11)-1);
dx(28)=+x(12)*(x(12)-1);
end
```

Co-funded by the
Erasmus+ Programme
of the European Union

MATLAB-CODING of the Neurocomputing concept

- N3 (Source) and N4 (Destination)

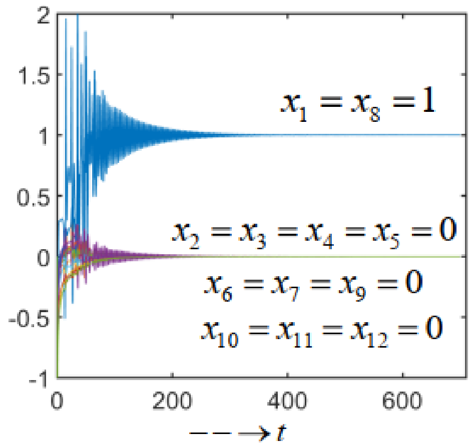


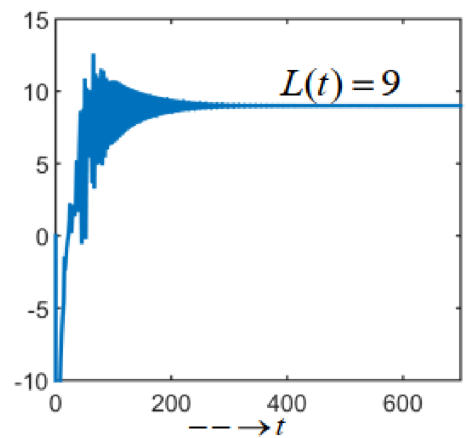



```
function dx=f(t,x)
dx=zeros(28,1);
c1=1;c2=2;c3=3;c4=4;c5=5;c6=6;c7=7;c8=8;c9=9;c10=10;
c11=11;c12=12;              s1=0; s2=0; s3=1; s4=-1;

dx(1)=-( c1 + x(13) - x(15) + x(17)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(15) + x(18)*(2*x(2)-1) );
dx(3)=-( c3 - x(14) + x(15) + x(19)*(2*x(3)-1) );
dx(4)=-( c4 + x(14) - x(15) + x(20)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(21)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) - x(14) + x(22)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) - x(16) + x(23)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(16) + x(24)*(2*x(8)-1) );
dx(9)=-( c9 - x(14) + x(16) + x(25)*(2*x(9)-1) );
dx(10)=-( c10 + x(14) - x(16) + x(26)*(2*x(10)-1) );
dx(11)=-( c11 - x(15) + x(16) + x(27)*(2*x(11)-1) );
dx(12)=-( c12 + x(15) - x(16) + x(28)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s2 );
dx(15)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s3 );
dx(16)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s4 );

dx(17)=+x(1)*(x(1)-1);
dx(18)=+x(2)*(x(2)-1);
dx(19)=+x(3)*(x(3)-1);
dx(20)=+x(4)*(x(4)-1);
dx(21)=+x(5)*(x(5)-1);
dx(22)=+x(6)*(x(6)-1);
dx(23)=+x(7)*(x(7)-1);
dx(24)=+x(8)*(x(8)-1);
dx(25)=+x(9)*(x(9)-1);
dx(26)=+x(10)*(x(10)-1);
dx(27)=+x(11)*(x(11)-1);
dx(28)=+x(12)*(x(12)-1);
end
```

MATLAB-CODING of the Neurocomputing concept

- N4 (Source) and N3 (Destination)






```
function dx=f(t,x)
dx=zeros(28,1);
c1=1;c2=2;c3=3;c4=4;c5=5;c6=6;c7=7;c8=8;c9=9;c10=10;
c11=11;c12=12;              s1=0; s2=0; s3=-1; s4=1;

dx(1)=-( c1 + x(13) - x(15) + x(17)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(15) + x(18)*(2*x(2)-1) );
dx(3)=-( c3 - x(14) + x(15) + x(19)*(2*x(3)-1) );
dx(4)=-( c4 + x(14) - x(15) + x(20)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(21)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) - x(14) + x(22)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) - x(16) + x(23)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(16) + x(24)*(2*x(8)-1) );
dx(9)=-( c9 - x(14) + x(16) + x(25)*(2*x(9)-1) );
dx(10)=-( c10 + x(14) - x(16) + x(26)*(2*x(10)-1) );
dx(11)=-( c11 - x(15) + x(16) + x(27)*(2*x(11)-1) );
dx(12)=-( c12 + x(15) - x(16) + x(28)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s2 );
dx(15)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s3 );
dx(16)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s4 );

dx(17)=+x(1)*(x(1)-1);
dx(18)=+x(2)*(x(2)-1);
dx(19)=+x(3)*(x(3)-1);
dx(20)=+x(4)*(x(4)-1);
dx(21)=+x(5)*(x(5)-1);
dx(22)=+x(6)*(x(6)-1);
dx(23)=+x(7)*(x(7)-1);
dx(24)=+x(8)*(x(8)-1);
dx(25)=+x(9)*(x(9)-1);
dx(26)=+x(10)*(x(10)-1);
dx(27)=+x(11)*(x(11)-1);
dx(28)=+x(12)*(x(12)-1);
end
```

CASE STUDY 3:

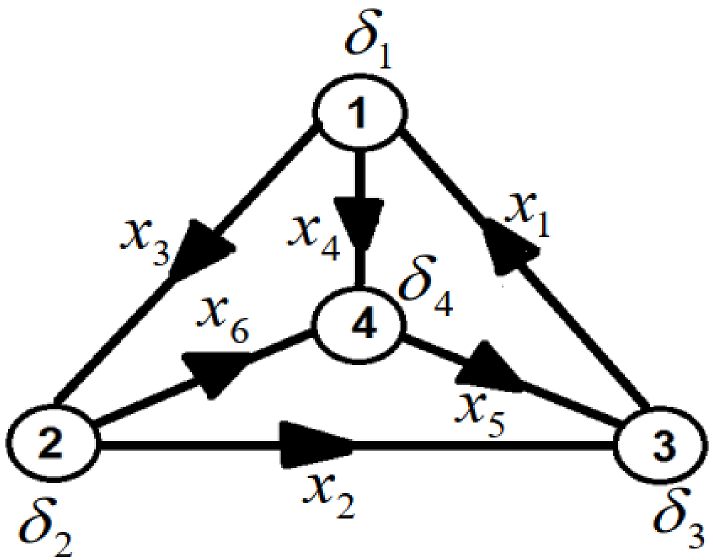
Traveler Salesman Problem (TSP)

„A directed graph of magnitude 4 and size 6“

Co-funded by the
Erasmus+ Programme
of the European Union

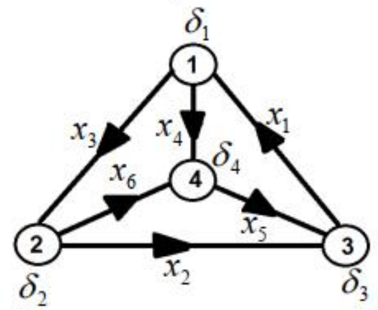
Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (1)

- Consider the directed graph of magnitude 4 and size 6 in Fig. 1:
  - ✓ Use the theory of optimization to model the TSP problem in Fig. 1 mathematically.



Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (2)

- Mathematical modeling of the TSP problem in Fig. 1.



Step 1: Expression of the total cost of the graph in Fig. 1 and objective function

$$Min\left[f(x_i, c_i) = \left(c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5 + c_6x_6\right)\right]$$

Step 2: Assignement of the three possible attributes to each node of Fig. 1

$$g(x_i,\delta_i) = \begin{cases} \text{Node1}: \ (x_3 + x_4 - x_1) = \delta_1 \ \cap \ (x_3 + x_4 + x_1) = \delta_2 \\ \text{Node2}: \ (x_2 + x_6 - x_3) = \delta_3 \ \cap \ (x_2 + x_6 + x_3) = \delta_4 \\ \text{Node3}: \ (x_1 - x_5 - x_2) = \delta_5 \ \cap \ (x_1 + x_5 + x_2) = \delta_6 \\ \text{Node4}: \ (x_5 - x_6 - x_4) = \delta_7 \ \cap \ (x_5 + x_6 + x_4) = \delta_8 \end{cases}$$

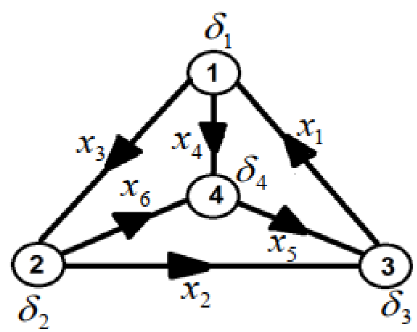
$$\begin{cases} \delta_i = 0 \ \ (i: \text{odd}) \\ \delta_i = 2 \ \ (i: \text{even}) \end{cases}$$

Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (3)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 3: Expression of constraints related to binarization

$$h(x_i)=\begin{cases} x_1(x_1-1)=0 \\ x_2(x_2-1)=0 \\ x_3(x_3-1)=0 \\ x_4(x_4-1)=0 \\ x_5(x_5-1)=0 \\ x_6(x_6-1)=0 \end{cases}$$



Step 4: Expression of the Lagrange function as the total energy of the system

$$L=(c_1x_1+c_2x_2+c_3x_3+c_4x_4+c_5x_5+c_6x_6)+\lambda_1(x_3+x_4-x_1-\delta_1)+\lambda_2(x_3+x_4+x_1-\delta_2)+\lambda_3(x_2+x_6-x_3-\delta_3)$$
$$+\lambda_4(x_2+x_6+x_3-\delta_4)+\lambda_5(x_1-x_5-x_2-\delta_5)+\lambda_6(x_1+x_5+x_2-\delta_6)+\lambda_7(x_5-x_6-x_4-\delta_7)+\lambda_8(x_5+x_6+x_4-\delta_8)$$
$$+\lambda_9x_1(x_1-1)+\lambda_{10}x_2(x_2-1)+\lambda_{11}x_3(x_3-1)+\lambda_{12}x_4(x_4-1)+\lambda_{13}x_5(x_5-1)+\lambda_{14}x_6(x_6-1)$$

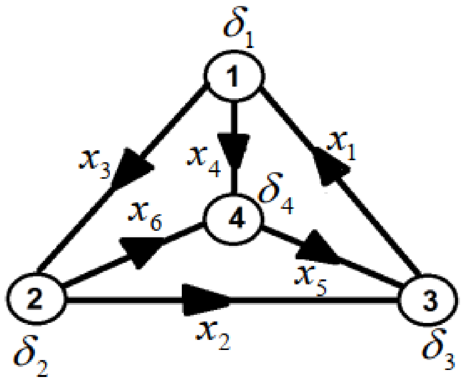
Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (4)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 5: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient descent to decision neurons

$$
\begin{cases}
\frac{dx_1}{dt} = -\frac{\partial L}{\partial x_1} = -[c_1 - \lambda_1 + \lambda_2 + \lambda_5 + \lambda_6 + \lambda_9(2x_1 - 1)] \\
\frac{dx_2}{dt} = -\frac{\partial L}{\partial x_2} = -[c_2 + \lambda_3 + \lambda_4 - \lambda_5 + \lambda_6 + \lambda_{10}(2x_2 - 1)] \\
\frac{dx_3}{dt} = -\frac{\partial L}{\partial x_3} = -[c_3 + \lambda_1 + \lambda_2 - \lambda_3 + \lambda_4 + \lambda_{11}(2x_3 - 1)] \\
\frac{dx_4}{dt} = -\frac{\partial L}{\partial x_4} = -[c_4 + \lambda_1 + \lambda_2 - \lambda_7 + \lambda_8 + \lambda_{12}(2x_4 - 1)] \\
\frac{dx_5}{dt} = -\frac{\partial L}{\partial x_5} = -[c_5 - \lambda_5 + \lambda_6 + \lambda_7 + \lambda_8 + \lambda_{13}(2x_5 - 1)] \\
\frac{dx_6}{dt} = -\frac{\partial L}{\partial x_6} = -[c_6 + \lambda_3 + \lambda_4 - \lambda_7 + \lambda_8 + \lambda_{14}(2x_6 - 1)]
\end{cases}
$$



Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (5)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 6: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 1)

$$\begin{cases} \dfrac{d\lambda_1}{dt} = +\dfrac{\partial L}{\partial \lambda_1} = +\left[x_3 + x_4 - x_1 - \delta_1\right] \\ \dfrac{d\lambda_2}{dt} = +\dfrac{\partial L}{\partial \lambda_2} = +\left[x_3 + x_4 + x_1 - \delta_2\right] \\ \dfrac{d\lambda_3}{dt} = +\dfrac{\partial L}{\partial \lambda_3} = +\left[x_2 + x_6 - x_3 - \delta_3\right] \\ \dfrac{d\lambda_4}{dt} = +\dfrac{\partial L}{\partial \lambda_4} = +\left[x_2 + x_6 + x_3 - \delta_4\right] \end{cases}
\qquad
\begin{cases} \dfrac{d\lambda_5}{dt} = +\dfrac{\partial L}{\partial \lambda_5} = +\left[x_1 - x_5 - x_2 - \delta_5\right] \\ \dfrac{d\lambda_6}{dt} = +\dfrac{\partial L}{\partial \lambda_6} = +\left[x_1 + x_5 + x_2 - \delta_6\right] \\ \dfrac{d\lambda_7}{dt} = +\dfrac{\partial L}{\partial \lambda_7} = +\left[x_5 - x_6 - x_4 - \delta_7\right] \\ \dfrac{d\lambda_8}{dt} = +\dfrac{\partial L}{\partial \lambda_8} = +\left[x_5 + x_6 + x_4 - \delta_8\right] \end{cases}$$

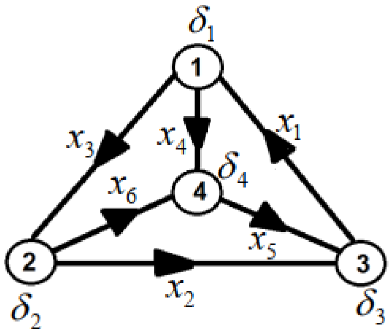
Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (6)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 7: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 2)

$$\begin{cases} \frac{d\lambda_9}{dt} = +\frac{\partial L}{\partial \lambda_9} = +\left[x_1\left(x_1 - 1\right)\right] \\ \frac{d\lambda_{10}}{dt} = +\frac{\partial L}{\partial \lambda_{10}} = +\left[x_2\left(x_2 - 1\right)\right] \\ \frac{d\lambda_{11}}{dt} = +\frac{\partial L}{\partial \lambda_{11}} = +\left[x_3\left(x_3 - 1\right)\right] \\ \frac{d\lambda_{12}}{dt} = +\frac{\partial L}{\partial \lambda_{12}} = +\left[x_4\left(x_4 - 1\right)\right] \\ \frac{d\lambda_{13}}{dt} = +\frac{\partial L}{\partial \lambda_{13}} = +\left[x_5\left(x_5 - 1\right)\right] \\ \frac{d\lambda_{14}}{dt} = +\frac{\partial L}{\partial \lambda_{14}} = +\left[x_6\left(x_6 - 1\right)\right] \end{cases}$$



Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (7)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 8: Neurocomputing concept- Derivation of the set of coupled ODEs model

Group 1 : Decision Neurons

$$\begin{cases} \frac{dx_1}{dt} = -\frac{\partial L}{\partial x_1} = -[c_1 - \lambda_1 + \lambda_2 + \lambda_5 + \lambda_6 + \lambda_9(2x_1 - 1)] \\ \frac{dx_2}{dt} = -\frac{\partial L}{\partial x_2} = -[c_2 + \lambda_3 + \lambda_4 - \lambda_5 + \lambda_6 + \lambda_{10}(2x_2 - 1)] \\ \frac{dx_3}{dt} = -\frac{\partial L}{\partial x_3} = -[c_3 + \lambda_1 + \lambda_2 - \lambda_3 + \lambda_4 + \lambda_{11}(2x_3 - 1)] \\ \frac{dx_4}{dt} = -\frac{\partial L}{\partial x_4} = -[c_4 + \lambda_1 + \lambda_2 - \lambda_7 + \lambda_8 + \lambda_{12}(2x_4 - 1)] \\ \frac{dx_5}{dt} = -\frac{\partial L}{\partial x_5} = -[c_5 - \lambda_5 + \lambda_6 + \lambda_7 + \lambda_8 + \lambda_{13}(2x_5 - 1)] \\ \frac{dx_6}{dt} = -\frac{\partial L}{\partial x_6} = -[c_6 + \lambda_3 + \lambda_4 - \lambda_7 + \lambda_8 + \lambda_{14}(2x_6 - 1)] \end{cases}$$

Group 2 : Multiplier Neurons

$$\begin{cases} \frac{d\lambda_1}{dt} = +\frac{\partial L}{\partial \lambda_1} = +[x_3 + x_4 - x_1 - \delta_1] \\ \frac{d\lambda_2}{dt} = +\frac{\partial L}{\partial \lambda_2} = +[x_3 + x_4 + x_1 - \delta_2] \\ \frac{d\lambda_3}{dt} = +\frac{\partial L}{\partial \lambda_3} = +[x_2 + x_6 - x_3 - \delta_3] \\ \frac{d\lambda_4}{dt} = +\frac{\partial L}{\partial \lambda_4} = +[x_2 + x_6 + x_3 - \delta_4] \\ \frac{d\lambda_5}{dt} = +\frac{\partial L}{\partial \lambda_5} = +[x_1 - x_5 - x_2 - \delta_5] \\ \frac{d\lambda_6}{dt} = +\frac{\partial L}{\partial \lambda_6} = +[x_1 + x_5 + x_2 - \delta_6] \\ \frac{d\lambda_7}{dt} = +\frac{\partial L}{\partial \lambda_7} = +[x_5 - x_6 - x_4 - \delta_7] \\ \frac{d\lambda_8}{dt} = +\frac{\partial L}{\partial \lambda_8} = +[x_5 + x_6 + x_4 - \delta_8] \end{cases}$$

Group 3 : Multiplier Neurons

$$\begin{cases} \frac{d\lambda_9}{dt} = +\frac{\partial L}{\partial \lambda_9} = +[x_1(x_1 - 1)] \\ \frac{d\lambda_{10}}{dt} = +\frac{\partial L}{\partial \lambda_{10}} = +[x_2(x_2 - 1)] \\ \frac{d\lambda_{11}}{dt} = +\frac{\partial L}{\partial \lambda_{11}} = +[x_3(x_3 - 1)] \\ \frac{d\lambda_{12}}{dt} = +\frac{\partial L}{\partial \lambda_{12}} = +[x_4(x_4 - 1)] \\ \frac{d\lambda_{13}}{dt} = +\frac{\partial L}{\partial \lambda_{13}} = +[x_5(x_5 - 1)] \\ \frac{d\lambda_{14}}{dt} = +\frac{\partial L}{\partial \lambda_{14}} = +[x_6(x_6 - 1)] \end{cases}$$

Mathematical modeling of the Traveler Salesman problem in a directed graph of magnitude 4 and size 6 (8)

- Mathematical modeling of the TSP problem in Fig. 1.

Step 8: Neurocomputing concept- Derivation of the set of cou

$$\lambda_1 = x_7 ; \lambda_2 = x_8 ; \lambda_3 = x_9 ; \lambda_4 = x_{10};$$
$$\lambda_5 = x_{11} ; \lambda_6 = x_{12}; \lambda_7 = x_{13} ; \lambda_8 = x_{14} ;$$
$$\lambda_9 = x_{15} ; \lambda_{10} = x_{16}; \lambda_{11} = x_{17} ; \lambda_{12} = x_{18} ;$$
$$\lambda_{13} = x_{19} ; \lambda_{14} = x_{20};$$

Group 1 : Decision Neurons

$$\begin{cases}
\frac{dx_1}{dt} = -\frac{\partial L}{\partial x_1} = -[c_1 - \lambda_1 + \lambda_2 + \lambda_5 + \lambda_6 + \lambda_9(2x_1 - 1)] \\
\frac{dx_2}{dt} = -\frac{\partial L}{\partial x_2} = -[c_2 + \lambda_3 + \lambda_4 - \lambda_5 + \lambda_6 + \lambda_{10}(2x_2 - 1)] \\
\frac{dx_3}{dt} = -\frac{\partial L}{\partial x_3} = -[c_3 + \lambda_1 + \lambda_2 - \lambda_3 + \lambda_4 + \lambda_{11}(2x_3 - 1)] \\
\frac{dx_4}{dt} = -\frac{\partial L}{\partial x_4} = -[c_4 + \lambda_1 + \lambda_2 - \lambda_7 + \lambda_8 + \lambda_{12}(2x_4 - 1)] \\
\frac{dx_5}{dt} = -\frac{\partial L}{\partial x_5} = -[c_5 - \lambda_5 + \lambda_6 + \lambda_7 + \lambda_8 + \lambda_{13}(2x_5 - 1)] \\
\frac{dx_6}{dt} = -\frac{\partial L}{\partial x_6} = -[c_6 + \lambda_3 + \lambda_4 - \lambda_7 + \lambda_8 + \lambda_{14}(2x_6 - 1)]
\end{cases}$$

Group 2 : Multiplier Neurons

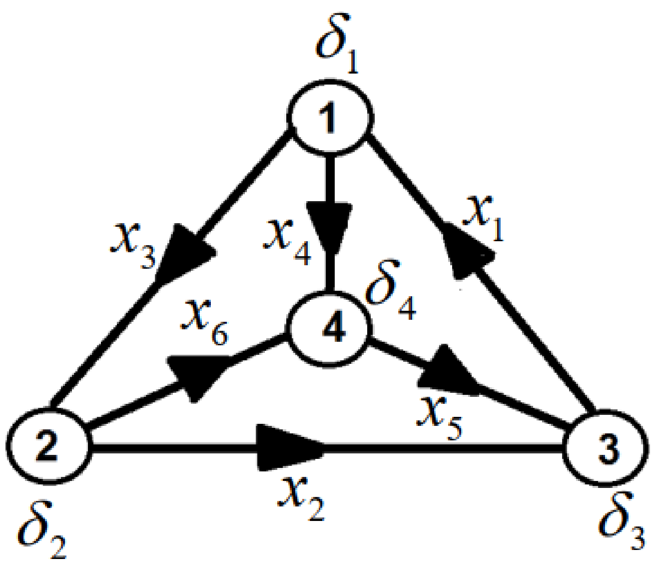
$$\begin{cases}
\frac{d\lambda_1}{dt} = +\frac{\partial L}{\partial \lambda_1} = +[x_3 + x_4 - x_1 - \delta_1] \\
\frac{d\lambda_2}{dt} = +\frac{\partial L}{\partial \lambda_2} = +[x_3 + x_4 + x_1 - \delta_2] \\
\frac{d\lambda_3}{dt} = +\frac{\partial L}{\partial \lambda_3} = +[x_2 + x_6 - x_3 - \delta_3] \\
\frac{d\lambda_4}{dt} = +\frac{\partial L}{\partial \lambda_4} = +[x_2 + x_6 + x_3 - \delta_4] \\
\frac{d\lambda_5}{dt} = +\frac{\partial L}{\partial \lambda_5} = +[x_1 - x_5 - x_2 - \delta_5] \\
\frac{d\lambda_6}{dt} = +\frac{\partial L}{\partial \lambda_6} = +[x_1 + x_5 + x_2 - \delta_6] \\
\frac{d\lambda_7}{dt} = +\frac{\partial L}{\partial \lambda_7} = +[x_5 - x_6 - x_4 - \delta_7] \\
\frac{d\lambda_8}{dt} = +\frac{\partial L}{\partial \lambda_8} = +[x_5 + x_6 + x_4 - \delta_8]
\end{cases}$$

Group 3 : Multiplier Neurons

$$\begin{cases}
\frac{d\lambda_9}{dt} = +\frac{\partial L}{\partial \lambda_9} = +[x_1(x_1 - 1)] \\
\frac{d\lambda_{10}}{dt} = +\frac{\partial L}{\partial \lambda_{10}} = +[x_2(x_2 - 1)] \\
\frac{d\lambda_{11}}{dt} = +\frac{\partial L}{\partial \lambda_{11}} = +[x_3(x_3 - 1)] \\
\frac{d\lambda_{12}}{dt} = +\frac{\partial L}{\partial \lambda_{12}} = +[x_4(x_4 - 1)] \\
\frac{d\lambda_{13}}{dt} = +\frac{\partial L}{\partial \lambda_{13}} = +[x_5(x_5 - 1)] \\
\frac{d\lambda_{14}}{dt} = +\frac{\partial L}{\partial \lambda_{14}} = +[x_6(x_6 - 1)]
\end{cases}$$

MATLAB-CODING of the Neurocomputing concept

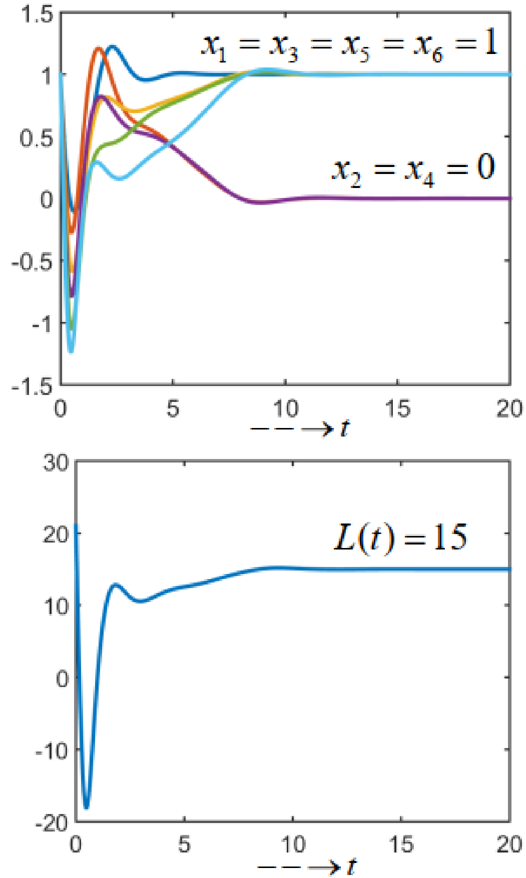
- Numerical simulation of the Traveler Salesman problem (TSP) in Fig. 1.




```
function dx=f(t,x)
dx=zeros(20,1);
c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=0; s2=2;   s3=0; s4=2;   s5=0; s6=2;   s7=0; s8=2;
dx(1)=-( c1 - x(7)  + x(8)  + x(11)+ x(12)+ x(15)*(2*x(1)-1) );
dx(2)=-( c2 + x(9)  + x(10)  - x(11)+ x(12)  + x(16)*(2*x(2)-1) );
dx(3)=-( c3 + x(7)  + x(8)  - x(9)+ x(10)+ x(17)*(2*x(3)-1) );
dx(4)=-( c4 + x(7)  + x(8)  - x(13)+ x(14)  + x(18)*(2*x(4)-1) );
dx(5)=-( c5 - x(11)  + x(12)  + x(13)+ x(14)+ x(19)*(2*x(5)-1) );
dx(6)=-( c6 + x(9)  + x(10)  - x(13)+ x(14)+ x(20)*(2*x(6)-1) );
dx(7)=+( x(3)  + x(4)  - x(1)  - s1 );
dx(8)=+( x(3)  + x(4)  + x(1)  - s2 );
dx(9)=+( x(6)  + x(2)  - x(3)  - s3 );
dx(10)=+( x(6)  + x(2)  + x(3)  - s4 );
dx(11)=+( x(1)  - x(2)  - x(5)  - s5 );
dx(12)=+( x(1)  + x(2)  + x(5)  - s6 );
dx(13)=+( x(5)  - x(4)  - x(6)  - s7 );
dx(14)=+( x(5)  + x(4)  + x(6)  - s8 );
dx(15)=+x(1)*(x(1)-1);
dx(16)=+x(2)*(x(2)-1);
dx(17)=+x(3)*(x(3)-1);
dx(18)=+x(4)*(x(4)-1);
dx(19)=+x(5)*(x(5)-1);
dx(20)=+x(6)*(x(6)-1);
end
```

MATLAB-CODING of the Neurocomputing concept

- Numerical simulation of the Traveler Salesman problem (TSP) in Fig. 1.




```
function dx=f(t,x)
dx=zeros(20,1);
c1=1; c2=2; c3=3; c4=4; c5=5; c6=6;
s1=0; s2=2;   s3=0; s4=2;   s5=0; s6=2;   s7=0; s8=2;
dx(1)=-( c1 - x(7) + x(8) + x(11)+ x(12)+ x(15)*(2*x(1)-1) );
dx(2)=-( c2 + x(9) + x(10) - x(11)+ x(12) + x(16)*(2*x(2)-1) );
dx(3)=-( c3 + x(7) + x(8) - x(9)+ x(10)+ x(17)*(2*x(3)-1) );
dx(4)=-( c4 + x(7) + x(8) - x(13)+ x(14) + x(18)*(2*x(4)-1) );
dx(5)=-( c5 - x(11) + x(12) + x(13)+ x(14)+ x(19)*(2*x(5)-1) );
dx(6)=-( c6 + x(9) + x(10) - x(13)+ x(14)+ x(20)*(2*x(6)-1) );
dx(7)=+( x(3) + x(4) - x(1) - s1 );
dx(8)=+( x(3) + x(4) + x(1) - s2 );
dx(9)=+( x(6) + x(2) - x(3) - s3 );
dx(10)=+( x(6) + x(2) + x(3) - s4 );
dx(11)=+( x(1) - x(2) - x(5) - s5 );
dx(12)=+( x(1) + x(2) + x(5) - s6 );
dx(13)=+( x(5) - x(4) - x(6) - s7 );
dx(14)=+( x(5) + x(4) + x(6) - s8 );
dx(15)=+x(1)*(x(1)-1);
dx(16)=+x(2)*(x(2)-1);
dx(17)=+x(3)*(x(3)-1);
dx(18)=+x(4)*(x(4)-1);
dx(19)=+x(5)*(x(5)-1);
dx(20)=+x(6)*(x(6)-1);

end
```

```
[t,x]=ode45(@TSP1, [0 20], [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ]);
```

CASE STUDY 4:

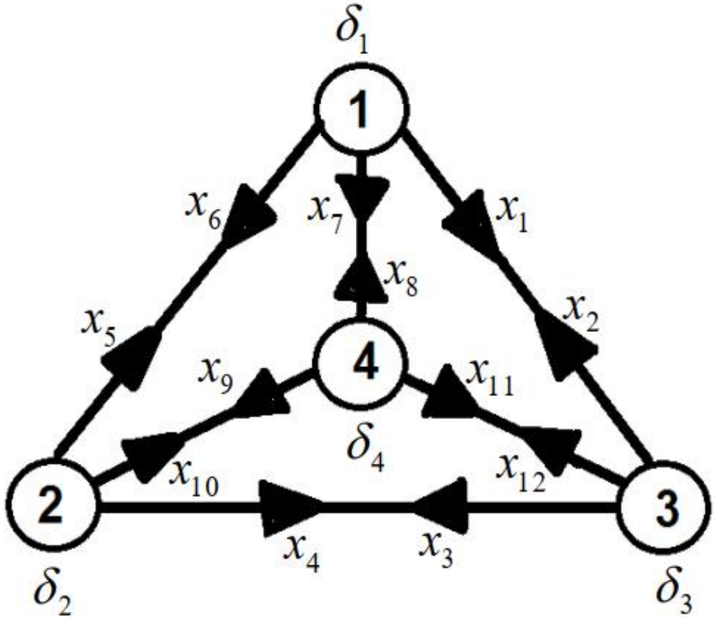
Traveler Salesman Problem (TSP)

„Undirected graph of magnitude 4 and size 12“

Co-funded by the Erasmus+ Programme of the European Union

Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (1)

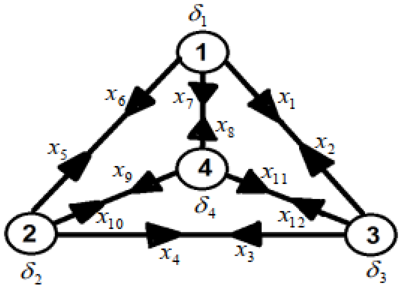
- Consider the undirected graph of magnitude 4 and size 12 in Fig. 2:
  - ✓ Use the theory of optimization to model the TSP problem in Fig. 2 mathematically.



Intelligent Transport Systems: New ICT – based Master's Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP

Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (2)

- Mathematical modeling of the TSP problem in Fig. 2.



Step 1: Expression of the total cost of the graph in Fig. 2 and objective function

$$Min\left[f(x_i, c_i) = c_1x_1 + c_2x_2 + c_3x_3 + c_4x_4 + c_5x_5 + c_6x_6 + c_7x_7 + c_8x_8 + c_9x_9 + c_{10}x_{10} + c_{11}x_{11} + c_{12}x_{12}\right]$$

Step 2: Assignement of the three possible attributes to each node of Fig. 2

$$g(x_i, \delta_i) = \begin{cases} \text{Node1}: (x_1 + x_6 + x_7 - x_2 - x_5 - x_8) = \delta_1 \cap (x_1 + x_6 + x_7 + x_2 + x_5 + x_8) = \delta_2 \\ \text{Node2}: (x_4 + x_5 + x_{10} - x_3 - x_6 - x_9) = \delta_3 \cap (x_4 + x_5 + x_{10} + x_3 + x_6 + x_9) = \delta_4 \\ \text{Node3}: (x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11}) = \delta_5 \cap (x_2 + x_3 + x_{12} + x_1 + x_4 + x_{11}) = \delta_6 \\ \text{Node4}: (x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12}) = \delta_7 \cap (x_8 + x_9 + x_{11} + x_7 + x_{10} + x_{12}) = \delta_8 \end{cases}$$

$$\begin{cases} \delta_i = 0 \ \ (i: \text{odd}) \\ \delta_i = 2 \ \ (i: \text{even}) \end{cases}$$

INTRAS

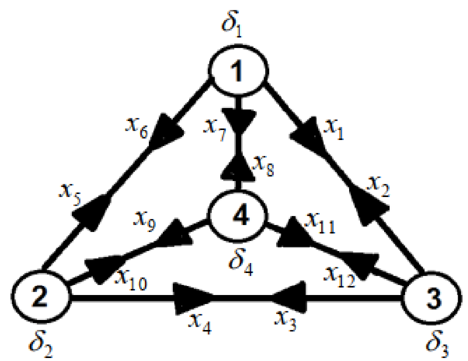
Co-funded by the Erasmus+ Programme of the European Union

Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (3)

- Mathematical modeling of the TSP problem in Fig. 2.

Step 3: Expression of constraints related to binarization

$$h(x_i)=\begin{cases} x_1(x_1-1)=0\\ x_2(x_2-1)=0\\ x_3(x_3-1)=0\\ x_4(x_4-1)=0\\ x_5(x_5-1)=0\\ x_6(x_6-1)=0 \end{cases} \qquad h(x_i)=\begin{cases} x_7(x_7-1)=0\\ x_8(x_8-1)=0\\ x_9(x_9-1)=0\\ x_{10}(x_{10}-1)=0\\ x_{11}(x_{11}-1)=0\\ x_{12}(x_{12}-1)=0 \end{cases}$$



Step 4: Expression of the Lagrange function as the total energy of the system

$$\begin{aligned} L=&(c_1x_1+c_2x_2+c_3x_3+c_4x_4+c_5x_5+c_6x_6+c_7x_7+c_8x_8+c_9x_9+c_{10}x_{10}+c_{11}x_{11}+c_{12}x_{12})+\lambda_1(x_1+x_6+x_7-x_2-x_5-x_8-\delta_1)\\ &+\lambda_2(x_1+x_6+x_7+x_2+x_5+x_8-\delta_2)+\lambda_3(x_4+x_5+x_{10}-x_3-x_6-x_9-\delta_3)+\lambda_4(x_4+x_5+x_{10}+x_3+x_6+x_9-\delta_4)\\ &+\lambda_5(x_2+x_3+x_{12}-x_1-x_4-x_{11}-\delta_5)+\lambda_6(x_2+x_3+x_{12}+x_1+x_4+x_{11}-\delta_6)+\lambda_7(x_8+x_9+x_{11}-x_7-x_{10}-x_{12}-\delta_7)\\ &+\lambda_8(x_8+x_9+x_{11}+x_7+x_{10}+x_{12}-\delta_8)+\lambda_9x_1(x_1-1)+\lambda_{10}x_2(x_2-1)+\lambda_{11}x_3(x_3-1)+\lambda_{12}x_4(x_4-1)+\lambda_{13}x_5(x_5-1)\\ &+\lambda_{14}x_6(x_6-1)+\lambda_{15}x_7(x_7-1)+\lambda_{16}x_8(x_8-1)+\lambda_{17}x_9(x_9-1)+\lambda_{18}x_{10}(x_{10}-1)+\lambda_{19}x_{11}(x_{11}-1)+\lambda_{20}x_{12}(x_{12}-1) \end{aligned}$$

Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (4)

- Mathematical modeling of the TSP problem in Fig. 2.

Step 5: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient descent to all twelve decision neurons

$$\begin{cases}
\dot{x}_1 = -[c_1 + x_{13} + x_{14} - x_{17} + x_{18} + x_{21}(2x_1 - 1)] \\
\dot{x}_2 = -[c_2 - x_{13} + x_{14} + x_{17} + x_{18} + x_{22}(2x_2 - 1)] \\
\dot{x}_3 = -[c_3 - x_{15} + x_{16} + x_{17} + x_{18} + x_{23}(2x_3 - 1)] \\
\dot{x}_4 = -[c_4 + x_{15} + x_{16} - x_{17} + x_{18} + x_{24}(2x_4 - 1)] \\
\dot{x}_5 = -[c_5 - x_{13} + x_{14} + x_{15} + x_{16} + x_{25}(2x_5 - 1)] \\
\dot{x}_6 = -[c_6 + x_{13} + x_{14} - x_{15} + x_{16} + x_{26}(2x_6 - 1)] \\
\dot{x}_7 = -[c_7 + x_{13} + x_{14} - x_{19} + x_{20} + x_{27}(2x_7 - 1)] \\
\dot{x}_8 = -[c_8 - x_{13} + x_{14} + x_{19} + x_{20} + x_{28}(2x_8 - 1)] \\
\dot{x}_9 = -[c_9 - x_{15} + x_{16} + x_{19} + x_{20} + x_{29}(2x_9 - 1)] \\
\dot{x}_{10} = -[c_{10} + x_{15} + x_{16} - x_{19} + x_{20} + x_{30}(2x_{10} - 1)] \\
\dot{x}_{11} = -[c_{11} - x_{17} + x_{18} + x_{19} + x_{20} + x_{31}(2x_{11} - 1)] \\
\dot{x}_{12} = -[c_{12} + x_{17} + x_{18} - x_{19} + x_{20} + x_{32}(2x_{12} - 1)]
\end{cases}$$

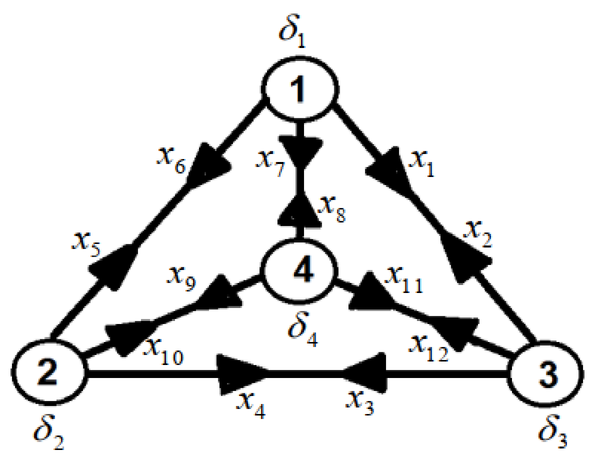
Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (5)

- Mathematical modeling of the TSP problem in Fig. 2.

Step 6: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 1)

$$
\begin{cases}
\dot{x}_{13} = \dot{\lambda}_1 = +[x_1 + x_6 + x_7 - x_2 - x_5 - x_8 - \delta_1] \\
\dot{x}_{14} = \dot{\lambda}_2 = +[x_1 + x_6 + x_7 + x_2 + x_5 + x_8 - \delta_2] \\
\dot{x}_{15} = \dot{\lambda}_3 = +[x_4 + x_5 + x_{10} - x_3 - x_6 - x_9 - \delta_3] \\
\dot{x}_{16} = \dot{\lambda}_4 = +[x_4 + x_5 + x_{10} + x_3 + x_6 + x_9 - \delta_4] \\
\dot{x}_{17} = \dot{\lambda}_5 = +[x_2 + x_3 + x_{12} - x_1 - x_4 - x_{11} - \delta_5] \\
\dot{x}_{18} = \dot{\lambda}_6 = +[x_2 + x_3 + x_{12} + x_1 + x_4 + x_{11} - \delta_6] \\
\dot{x}_{19} = \dot{\lambda}_7 = +[x_8 + x_9 + x_{11} - x_7 - x_{10} - x_{12} - \delta_7] \\
\dot{x}_{20} = \dot{\lambda}_8 = +[x_8 + x_9 + x_{11} + x_7 + x_{10} + x_{12} - \delta_8]
\end{cases}
$$



Mathematical modeling of the Traveler Salesman problem in an undirected graph of magnitude 4 and size 12 (6)

- Mathematical modeling of the TSP problem in Fig. 2.

Step 7: Application of the BDMM concept and derivation of the coupled ODEs

Application of gradient ascent to multiplier neurons (Group 2)

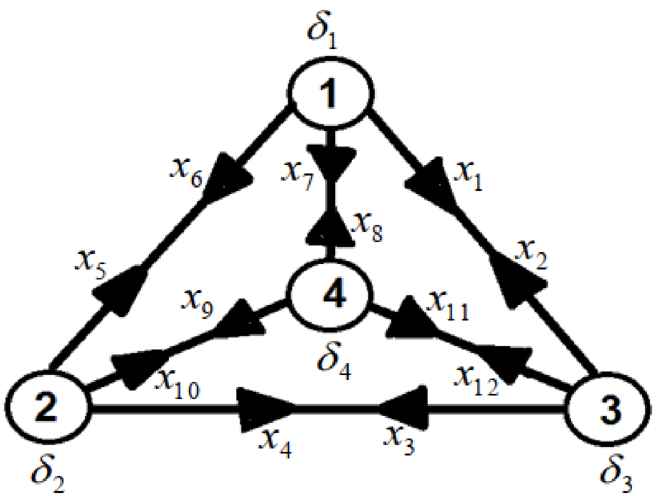
$$\begin{cases} \dot{x}_{21} = \dot{\lambda}_9 = +[x_1(x_1-1)] \\ \dot{x}_{22} = \dot{\lambda}_{10} = +[x_2(x_2-1)] \\ \dot{x}_{23} = \dot{\lambda}_{11} = +[x_3(x_3-1)] \\ \dot{x}_{24} = \dot{\lambda}_{12} = +[x_4(x_4-1)] \\ \dot{x}_{25} = \dot{\lambda}_{13} = +[x_5(x_5-1)] \\ \dot{x}_{26} = \dot{\lambda}_{14} = +[x_6(x_6-1)] \end{cases} \qquad \begin{cases} \dot{x}_{27} = \dot{\lambda}_{15} = +[x_7(x_7-1)] \\ \dot{x}_{28} = \dot{\lambda}_{16} = +[x_8(x_8-1)] \\ \dot{x}_{29} = \dot{\lambda}_{17} = +[x_9(x_9-1)] \\ \dot{x}_{30} = \dot{\lambda}_{18} = +[x_{10}(x_{10}-1)] \\ \dot{x}_{31} = \dot{\lambda}_{19} = +[x_{11}(x_{11}-1)] \\ \dot{x}_{32} = \dot{\lambda}_{20} = +[x_{12}(x_{12}-1)] \end{cases}$$

INTRAS

Co-funded by the
Erasmus+ Programme
of the European Union

MATLAB-CODING of the Neurocomputing concept

- Numerical simulation of the TSP

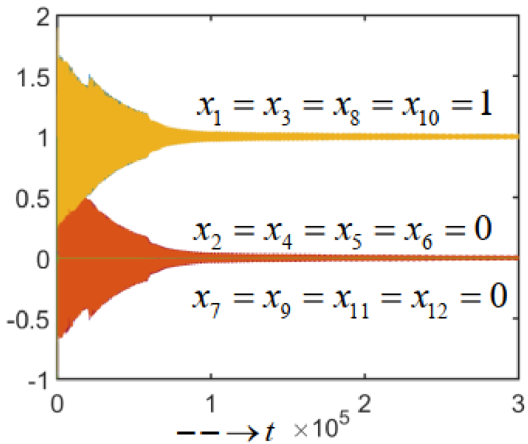


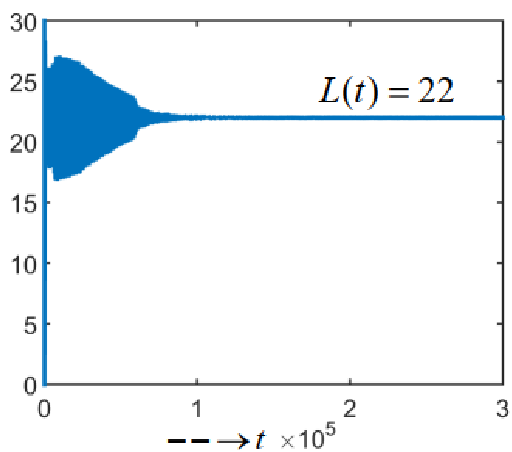

```
function dx=f(t,x)
 dx=zeros(32,1);
c1=1; c2=2; c3=3; c4=4; c5=5; c6=6; c7=7; c8=8; c9=9; c10=10; c11=11;
s1=0; s2=2;   s3=0; s4=2;   s5=0; s6=2;   s7=0; s8=2;         c12=12;
dx(1)=-( c1 + x(13) + x(14) - x(17) + x(18) + x(21)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(14) + x(17) + x(18) + x(22)*(2*x(2)-1) );
dx(3)=-( c3 - x(15) + x(16) + x(17) + x(18) + x(23)*(2*x(3)-1) );
dx(4)=-( c4 + x(15) + x(16) - x(17) + x(18) + x(24)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(15) + x(16) + x(25)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) + x(14) - x(15) + x(16) + x(26)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) + x(14) - x(19) + x(20) + x(27)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(14) + x(19) + x(20) + x(28)*(2*x(8)-1) );
dx(9)=-( c9 - x(15) + x(16) + x(19) + x(20) + x(29)*(2*x(9)-1) );
dx(10)=-( c10 + x(15) + x(16) - x(19) + x(20) + x(30)*(2*x(10)-1) );
dx(11)=-( c11 - x(17) + x(18) + x(19) + x(20) + x(31)*(2*x(11)-1) );
dx(12)=-( c12 + x(17) + x(18) - x(19) + x(20) + x(32)*(2*x(12)-1) );
dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(1) + x(6) + x(7) + x(2) + x(5) + x(8) - s2 );
dx(15)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s3 );
dx(16)=+( x(4) + x(5) + x(10) + x(3) + x(6) + x(9) - s4 );
dx(17)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s5 );
dx(18)=+( x(2) + x(3) + x(12) + x(1) + x(4) + x(11) - s6 );
dx(19)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s7 );
dx(20)=+( x(8) + x(9) + x(11) + x(7) + x(10) + x(12) - s8 );
dx(21)=+x(1)*(x(1)-1);
dx(22)=+x(2)*(x(2)-1);
dx(23)=+x(3)*(x(3)-1);
dx(24)=+x(4)*(x(4)-1);
dx(25)=+x(5)*(x(5)-1);
dx(26)=+x(6)*(x(6)-1);
dx(27)=+x(7)*(x(7)-1);
dx(28)=+x(8)*(x(8)-1);
dx(29)=+x(9)*(x(9)-1);
dx(30)=+x(10)*(x(10)-1);
dx(31)=+x(11)*(x(11)-1);
dx(32)=+x(12)*(x(12)-1);
end
```

UNIVERSITÄT
KLAGENFURT

MATLAB-CODING of the Neurocomputing concept

- Numerical simulation of the TSP






```
function dx=f(t,x)
dx=zeros(32,1);
c1=1; c2=2; c3=3; c4=4; c5=5; c6=6; c7=7; c8=8; c9=9; c10=10; c11=11;
s1=0; s2=2;  s3=0; s4=2;  s5=0; s6=2;  s7=0; s8=2;        c12=12;

dx(1)=-( c1 + x(13) + x(14) - x(17) + x(18) + x(21)*(2*x(1)-1) );
dx(2)=-( c2 - x(13) + x(14) + x(17) + x(18) + x(22)*(2*x(2)-1) );
dx(3)=-( c3 - x(15) + x(16) + x(17) + x(18) + x(23)*(2*x(3)-1) );
dx(4)=-( c4 + x(15) + x(16) - x(17) + x(18) + x(24)*(2*x(4)-1) );
dx(5)=-( c5 - x(13) + x(14) + x(15) + x(16) + x(25)*(2*x(5)-1) );
dx(6)=-( c6 + x(13) + x(14) - x(15) + x(16) + x(26)*(2*x(6)-1) );
dx(7)=-( c7 + x(13) + x(14) - x(19) + x(20) + x(27)*(2*x(7)-1) );
dx(8)=-( c8 - x(13) + x(14) + x(19) + x(20) + x(28)*(2*x(8)-1) );
dx(9)=-( c9 - x(15) + x(16) + x(19) + x(20) + x(29)*(2*x(9)-1) );
dx(10)=-( c10 + x(15) + x(16) - x(19) + x(20) + x(30)*(2*x(10)-1) );
dx(11)=-( c11 - x(17) + x(18) + x(19) + x(20) + x(31)*(2*x(11)-1) );
dx(12)=-( c12 + x(17) + x(18) - x(19) + x(20) + x(32)*(2*x(12)-1) );

dx(13)=+( x(1) + x(6) + x(7) - x(2) - x(5) - x(8) - s1 );
dx(14)=+( x(1) + x(6) + x(7) + x(2) + x(5) + x(8) - s2 );
dx(15)=+( x(4) + x(5) + x(10) - x(3) - x(6) - x(9) - s3 );
dx(16)=+( x(4) + x(5) + x(10) + x(3) + x(6) + x(9) - s4 );
dx(17)=+( x(2) + x(3) + x(12) - x(1) - x(4) - x(11) - s5 );
dx(18)=+( x(2) + x(3) + x(12) + x(1) + x(4) + x(11) - s6 );
dx(19)=+( x(8) + x(9) + x(11) - x(7) - x(10) - x(12) - s7 );
dx(20)=+( x(8) + x(9) + x(11) + x(7) + x(10) + x(12) - s8 );

dx(21)=+x(1)*(x(1)-1);
dx(22)=+x(2)*(x(2)-1);
dx(23)=+x(3)*(x(3)-1);
dx(24)=+x(4)*(x(4)-1);
dx(25)=+x(5)*(x(5)-1);
dx(26)=+x(6)*(x(6)-1);
dx(27)=+x(7)*(x(7)-1);
dx(28)=+x(8)*(x(8)-1);
dx(29)=+x(9)*(x(9)-1);
dx(30)=+x(10)*(x(10)-1);
dx(31)=+x(11)*(x(11)-1);
dx(32)=+x(12)*(x(12)-1);

end
```

```
[t,x]=ode45(@TSP2, [0 300000], [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]);
```

Co-funded by the Erasmus+ Programme of the European Union

CONCLUDING REMARKS

- We have demonstrated why is mathematics so important in transportation
- We have presented some/selected concret (or real-life) application examples of mathematics in transportation
- We have described the important information/knowledge to be provided (in each chapter) by the Lecturer
- We have selected some projects to be considered (by the Lecturer) in the frame of this Course/Lecture.
- **As a didactic example**, we have selected amongst the eight chapters of this Course the chapter entitled **„Mathematical modeling of graph theoretical problems- SPP & TSP“.** The full content of this chapter has been presented.

Co-funded by the
Erasmus+ Programme
of the European Union

END

UNIVERSITÄT KLAGENFURT

Intelligent Transport Systems: New ICT – based Master's Curricula in Uzbekistan (INTRAS)
Agreement number: 2017-3516/001-001
Project reference number: 586292-EPP-1-2017-1-PL-EPPKA2-CBHE-JP


Thank You


Q!