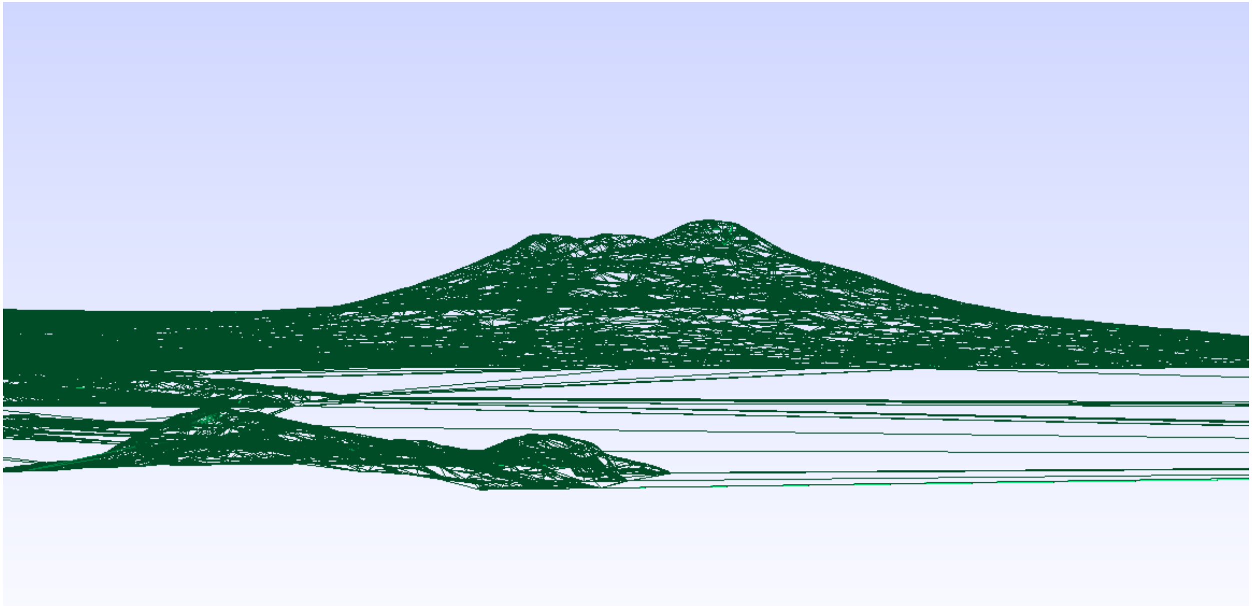
Computer aided design of a water supply network



Cours : PROJ_P2019 - Innovation project
Groupe : P5C006AJ
Responsable pédagogique :
BARBARULO *Andrea,* GATTI *Filippo*
Version : 1.0 du *22/01/18*

Auteurs :
CALVET-LITZELL Pere
COGNETTI Giovanni
MARZORATI Edoardo
VACCARINO Marcello

Abstract

Water supply remains a major issue in several countries. When designing a water-supply network optimality is a priority. The aim of this project is to find optimal network structures using automation and machine learning.

The development process is divided into several stages. During the first stage, network topology has been studied. A network has been designed using our software on real-world data. Next stages will involve adding further parameters such as water velocity or pressure to the existing model.

The project has a multidisciplinary nature. Using geographical data requires a certain level of acquaintance with different formats and software such as QGis. On the other hand, mastering a programming language like Python is required to implement the different algorithms and libraries.

Contents

Chapter 1

Introduction

Water supply remains one of the major issues in developing countries. Through this project we seek to improve the design of water supply networks. Designing water supply networks is an optimization problem in which engineers have to find the best balance between cost, transport efficiency and resistance to failure.

Biologically inspired models can provide interesting insights. Organisms that have gone through several rounds of evolutionary selection seem to be able to deliver efficient and nearly-optimal solutions. The use of such models seems to have produced satisfactory results for transport networks.

On the other hand, given the problem's complexity, the use of Artificial Neural Networks may help to find an optimal solution taking into account a large number of constraints. The water supply problem can be seen as a routing problem. This has been studied in several papers. Finally, Artificial Neural Networks could also be used in water resources management. Their potential in forecasting has been discussed. They could allow creating a water network that adapts itself to environmental conditions such as rain or drought.

Several papers strongly advise using neural networks for all these purposes.

1.1 Root system architecture

Reading Chloé Arson's presentation on bio-inspired geomechanics, we discovered the potential advantage of using root system architecture to design water lines. Prof. Arson conducted an experiment to compare the predictions of a root growth model with real water line networks. Root growth is a gene-controlled phenomenon. Therefore, different species may present different growth patterns. In addition, soil structure has also an influence on root structures. For example the presence of physical obstacles, such as boulders, alters geotropic growth. Prof. Arson also pointed out that a rocky soil would require a different model. Other characteristics like water and nutrient gradients or bacteria play a key role in root growth. Prof. Arson's experiment consisted in growing roots on a scale plastic model of the Georgia Tech campus. The results would allow to validate the accuracy of the mathematical model. Afterwards they could be compared to the existing water network and thus assess its efficiency. Prof. Arson also introduced leaf venation systems which bear certain resemblance to water line networks. Indeed, the growth of a leaf is governed by the presence of auxin (plant hormones) sources which can be seen as the nutrient sources of the root model.

We contacted Prof. Arson who gave us a very interesting bibliography on the subject of root growth models. Prof. Pierret's article stresses the complex relationship between soil structure and soil biological activity. Soil is a habitat for many organisms and is also responsible for the movement and transport of resources which are necessary for their survival. Through their roots, plants play a key role in many soil processes. Soil properties affect root growth which in turn affects resource acquisition and therefore the plant's impact on its environment (soil). Interest for root systems architecture comes from the necessity in agriculture of increasing productivity and minimizing water and nutrient losses. A good understanding of soil processes seems necessary to achieve this end. Moreover, Pierret points out that whereas soil biological and chemical processes have been carefully studied, physical processes need more attention. The article examines main biological factors that influence soil processes. It underlines the complex interactions between physical and chemical-biological processes and the impossibility to treat them separately. According to Pierret, roots are essential to study this complexity. In the second part of the article, the huge diversity of root classes is examined. This implies the necessity of using specific models for each species. The last part of the article discusses how modelling can provide clearer insights on the interactions between roots and soil.

Lionel Dupuy's article describes the evolution of root growth models. The first models appeared in the early 1970s and focused mainly on root length. However since the 1990s new complex models have emerged thanks to the use of more powerful computers. This phenomenon has been fostered by the "need for predictive technologies" at different scales. Dupuy suggests a new theoretical framework which takes into account individual root developmental parameters. He introduces "equations in discretized domains that deform as a result of growth". Simulations conducted by Dupuy have revealed some patterns in what seemed a complex and heterogeneous problem. More precisely, it seems that roots develop following travelling wave patterns of meristems.

V. M. Dunbabin also mentions the progress accomplished in the area of root growth modelling. The early models did not take into account the root growth in response to a heterogeneous soil environment. Nowadays, models must include soil properties and accurate descriptions of plant function. The aim of these simulations is again to provide a better understanding of the efficient acquisition of water and nutrients by plants. Resource availability has a clear impact on both the roots and the stem of the plan. For example, a low nutrient concentration diminishes shoot growth and therefore leaf and stem mass fractions as well. It has been observed that roots respond locally to soil properties. This characteristic allows the plant to forage with more precision and reduce metabolic cost. Three-dimensional models are able to seize the complexity of the problem. Previous models were rather simple and relied upon one-dimensional functions of rooting depth vs. time.

One of the most interesting articles is Atsushi Tero's "Rules for Biologically Inspired Adaptive Network Design". In order to solve the problem of transport networks efficiency, Tero created a mathematical model based on organisms that build biological networks. He explains that these biological networks have been honed by many rounds of evolutionary selection and that they can provide inspiration to design new networks. He praises their good balance between cost, transport efficiency and, above all, fault tolerance. One of such organisms is physarum polycephalum, a type of slime mold. Tero let physarum grow on a map of the Tokyo area where major cities were marked by food sources. A first network was obtained. In order to improve the results, the experiment was carried out a second time. However, illumination was used to introduce the real geographical constraints such as coastlines or mountains(illumination reduces physarum's growth). The results were very satisfactory and the biological network was very similar to the existing Tokyo transport network. Tero developed a mathematical model that tried to reproduce Physarum's behavior. The principle of the model is that tube thickness depends on the internal flow of nutrients. Thus a high rate tends thickens a tube and a low rate leads to its decay. As shown by Prof. Arsons' paper "Bio-inspired fluid extraction model for reservoir rocks", slime mold growth can also be used to study the flow in a porous medium.

Chapter 2

Algorithm design

2.1 Artificial intelligence application to routing problems

The growing of network usage and their increasing complexity, in particular for communication technologies application, drives towards the improvement of routing technique. One track of this research is the development of "intelligent" techniques for network design and management.

For our project we chose to follow this direction, combine sub-optimal AI algorithms to develop a possibly innovative solution. Lead by example, we will give an overview of the most edge braking applications in this field. This will allow us to introduce the main concepts and get down to the techniques we focused on.

AI are applied to many complex routing problems: one example is very large-scale integration (VLSI). The process of designing integrated circuits is hard due to the large number of often conflicting factors that affect the routing quality such as minimum area, wire length. Rostam Joobbani from Carnegie-Mellon University (1986), proved that an AI approach to the subject, a knowledge-based routing expert, could dramatically improve performances.

A more recent example is the use of AI in sensor wireless network. WSNs are spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to other locations. Management of those networks is particularly challenging because of the dynamic environmental conditions. J. Barbancho and al. (2007) wrote a review about the use of artificial intelligence techniques for WSNs for path discovery and other purposes. The study shows the potential of Artificial Neural Networks.

Artificial Neural Network learn to do tasks by considering examples, generally without task-specific programming. An ANN is based on a collection of connected units called artificial neurons. Each connection (synapse) between neurons can transmit a signal to another neuron. The receiving (postsynaptic) neuron can process the signal(s) and then signal downstream neurons connected to it.

N. Ahad,J. Quadir, N. Ahsan (2016) published a review focused on techniques and applications of artificial neural networks for wireless networks. The advantage of using ANN is that can make the network adaptive and able to predict user demand.

Concerning shortest path problems Michael Turcanik (2012) used an Hopfield neural network as a content-addressable memory for routing table look-up. A routing table is a database that keeps track of paths in a network. Whenever a node needs to send data to another node on a network, it must first know where to send it. If the node cannot directly connect to the destination node, it has to send it via other nodes along a proper route to the destination node. Most nodes do not try to figure out which route(s) might work; instead, a node will send the message to a gateway in the local area network, which then decides how to route the "package" of data to the correct destination. Each gateway will need to keep track of which way to deliver various packages of data, and for this it uses a Routing Table. Turcanik replaced the table with an ANN. His study shows the performance of routing table look-up in terms of speed and adaptability.

This excursus gives an idea of the incredibly various application of ANN in routing problems. We would like now to focus on the use of Hopfield Neural Network, which is the most classical solution for routing problems with ANN's.

Hopfield (1984) proposed the use of his algorithm to give heuristic solutions to the travel sales man problem. TSP is a well known NP-hard minimization problem. As defined by Karl Menger the TSP is "the task to find, for finitely many points whose pairwise distances are known, the shortest route connecting the points". So having n cities, our travel sales-man have to associate to each city X a position k in the tour so that:

$$\sum_{X}\sum_{Y\neq X}\sum_{j} d_{XY}y_{Xj}(y_{Y,j+1}+y_{Y,j-1})$$

is minimal, where dXY is the distance between city X and Y.

E. Wacholder and al. (1989) developed a more efficient implementation of the Hopfield NN for the travel sales-man problem. The algorithm was successfully tested on many problems with up to 30 cities and five salesmen, while a non-optimized brute-force approach would take billions of billions of years to return. In all test cases, the algorithm always converged to valid solutions.

Mustafa K. Mehmet Ali and Faouzi Kamoun (1993) considered modeling shortest path problem with Hopfield Neural Network for the first time. The researchers asserted that HNN can find shortest path effectively and sometimes it would be better to use such a network instead of classic algorithms such as Dijkstra.

We will know explain what Hopfield Neural Networks, with particular attention to the TSP application, although the definition we will give are general. Is a recurrent ANN, as opposed to feed forward NN, which means neurons interconnections forms a directed cycle, so neurons are both input and output. Hopfield nets are sets n2 nodes where X [1, n] and k [1, n] and the state is characterized by the binary activation values y = (yXj) of the nodes. A TSP problem with n cities can be modeled as an Hopfield net of dimension n2, where yXj is 1 if the city X is in the k-position of the tour.

The input sk(t+1) of the neuron k is:

$$s_k(t+1) = \sum_{j \neq k} y_i(t)\, w_{jk} + \theta_k$$

where wjk is the weight of the connection between j and k and is the bias The forward function is applied to the node input to obtain the new activation value at time t+1:

$$y_k(t) = sgn(s_k(t-1))$$

The energy function is as follow so that the optimal solution will minimize it:

$$E = \frac{A}{2}\sum_X \sum_j \sum_{k \neq j} y_{Xj} y_{Xk} + \frac{B}{2}\sum_j \sum_X \sum_{X \neq Y} y_{Xj} y_{Yj} + \frac{C}{2}\Big(\sum_X \sum_j y_{Xj} - n\Big)^2 + \frac{D}{2}\sum_X \sum_Y \sum_j d_{XY} y_{Xj}(y_{Y,j+1} + y_{Y,j-1})$$

The first two terms are null if and only if the there is a maximum of one active neuron for each row and column respectively. The third term is null if and only if there are n active neurons. The last term takes in account the distance of the path, that should be minimized as well.

The Hebbian rule to update the weights is deduced from the energy function:

$$w_{Xj,Yk} = -A\delta_{XY}\left(1 - \delta_{jK}\right) - B\delta_{jk}(1 - \delta_{XY}) - C - Dd_{XY}(\delta_{k,j+1} + \delta_{k,j-1})$$

where kj = 1 if j = k and zero otherwise. As in the energy function the first term inhibits connection within each row, the second within columns, the third is the global inhibition and the last term takes into account the distance between the cities.

Under the hypothesis $w_{Xj,Yk} = w_{Yk,Xj}$ the method can be proved to have stable points. At each iteration the net updates his parameters according to the Hebbian rule and the evolution of the state can be proved to be monotonically nonincreasing with respect of the energy function. Performing then a gradient descent, after a certain number of repetition the state converge to a stable point that is a minima of the energy function.

2.2 Artificial intelligence in water resources management

An other field of application, which is interesting for our work is water-resources management. While conventional linear regression models have been applied for stream-flow forecasting since 1970, AI has exhibited significant progress in forecasting and modeling non-linear hydrological applications. ANN's are used as statistical models for parameter estimation such forecasting of flow (including rainfall, streamflow and reservoir inflows), water quality variables such as algal concentrations

Holger R. Maier, Graeme C. Dandy provides an interesting insight in the method while M. R. Mustafa and al. (2012) published a review on the subject with particular focus on river sediment and discharge.

2.3 Design automation

The second part is the aqueduct design automation. In the set of nodes of the graph representing the topology there will be at least one water sinks and one source. The aim is to link sinks and sources in the best way possible.

Complexity comes out in the definition of the quality of the net as the factors that makes an aqueduct the optimal one are not easily modeled. We can suppose that variables such length, height, water speed and pressure, viscosity ecc should be taken into account. For our first approach, we decided to consider only the pipeline length so that we simplify acqueduct design to a classical routing problem. On this base, we will then be able to add complexity.

At this first stage, we want to find the optimal recovering-graph on the mesh graph connecting sinks and sources. For this end another graph is created and initialized with the buildings (sinks) and water sources (sources) nodes of the mesh graph.

2.4 Clustering

Running classical algorithms such a brute-force TSP or a minimum spanning tree to link the nodes in the sink-source graph would not be feasible for computational reasons. To break down the computational complexity of this operation we divide the aqueduct system in two layers: adduction and distribution nets. The adduction layer brings water from the source to the inhabited areas whereas the distribution segment is in charge of the "last kilometer" distribution. This two layer solution is commonly used in aqueduct design and network design in general: internet is an example. The advantages of this solutions are not only computational. Once the two layers are identified we can use different strategies to connect the nodes.

To find the inhabited areas we run the mean shift clustering algorithm on the sink-source graph. The implementation we use is the one in scikit-learn. Scikit-learn is a well-known machine learning library for Python and it features various classification, regression and clustering algorithms. [TODO spiegazione clustering] After this operation sink nodes are dived into clusters.

This operation is broken down in two tasks. First find all the path connecting sink and sources. Lets consider the set of sinks and sources, this is a sub-set of nodes of the mesh graph. On this set of nodes we create the complete graph. Edges of this new graph are paths on the original mesh, so are subgraph themselves, and are find with an optimal approach, such as Floyd-Warshal algorithm. On this subgraph, we solve the TSP with an Hopfield neural network. This gives shortest net connecting all sinks and sources.

The results can than be printed using NetworkX draw tools or the Visualization Toolkit. The Visualization Toolkit (VTK) is an open-source software system for 3D computer graphics, image processing, and visualization. VTK interfaces with python and packages such PyVTK provides tools for manipulating VTK, for example reading and writing operations.

2.5 Routing

Know we can design the water systems connecting the sinks. Let's first consider the distribution layer, which is to say the problem of connecting the sinks of a cluster.
The solution will be a recovering graph of the mesh graph which connects all the sinks in the clusters. To solve it we first defined the complete graph connecting all the sinks and than calculating the minimum spanning tree. Consider that edges of this new graph are paths on the original mesh, so are subgraph themselves, and are find with an optimal approach, such as the Dijkstra algorithm.

The same approach can be used for the adduction system. The result is in the image: softwere correctly recognized the fraction and connected them with the network.

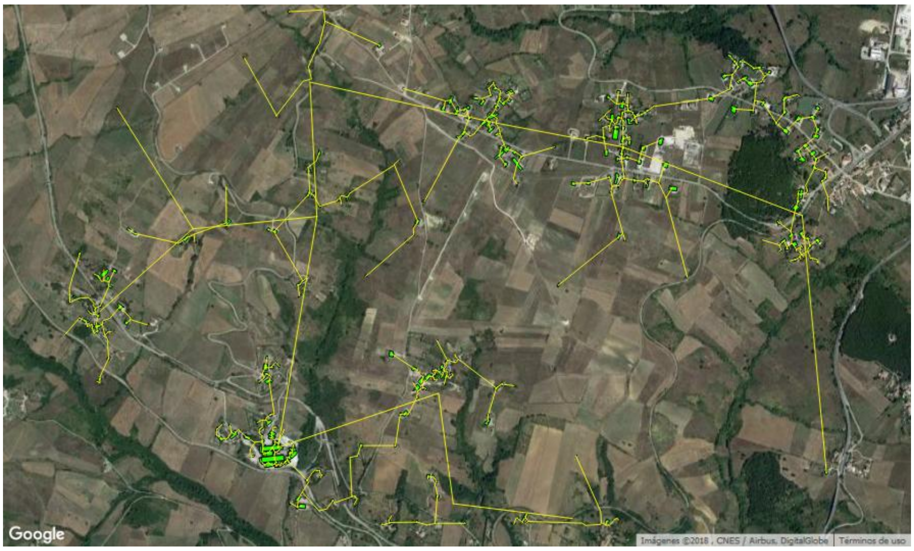


Figure 2.1: Calculated network surposed with satellite images, Limitone (near Potenza), Basilicata, Italy

Chapter 3

Data manipulation

3.1 Geographical Data

The overall idea is to take maps and automatically trace an aqueduct on it, in order to do that, we start from the map's shapefile. Shapefile is a popular geospatial vector data format for geographic information systems software. It spatially describes geometries: points, polylines and polygons. These, for example, could represent water wells, roads or buildings.

As those primitive geometrical data types come without any attributes to specify what they represent, a table of records to store attributes is provided. Websites like osm2shp or Geofabrik provide an immense database of shapefiles available for download. Moreover desktop software like Qgis provides shapefile editing tools. This way we can both download real-world maps and create our own.

Then through Qgis' meshing plug-in Gmsh we can mesh the surfaces of the map and export the result in vtk format as seen in Fig. 3.1 However, shapefiles seldom have information on the elevation (that is the Z coordinate) of the objects they represent. It is therefore necessary to use another format: the Digital Elevation Model (DEM). Digital Elevation Models provide this missing piece of information that can subsequently be added to the shapefile's attribute table.

DEMs can be converted into meshes thanks to software such as SAGA. Meshes saved as vtk files can easily be used in Python.

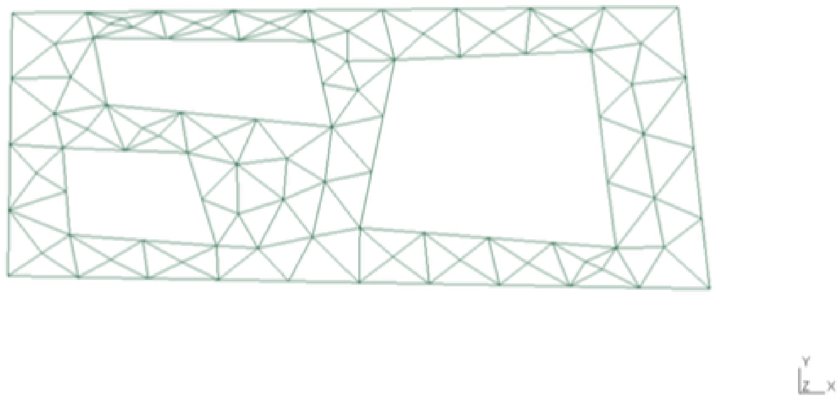


Figure 3.1: A simple mesh used for the software test

Indeed, vtk files are a simple and efficient way to describe mesh-like data structures. The vtk file boils down to those two elements: points and cells. Points have 3D coordinates while cells are surfaces, expressed by the points delimiting them. Point and cell data (scalar or vector) can also be assigned. We have therefore a file representing a graph, a classical mathematical model on which many operations can be performed: routing and clustering among others.

We now come to our software. Python has been chosen as easy to use, widespread programming language, good for rapid prototyping and rich in package and libraries. The problem is divided in two main tasks: modelling the data structure that represents the graph and the algorithmic part, the aqueduct design.

3.2 Data structure

To implement the data-structure we chose to use NetworkX. NetworkX is a Python package for the creation, manipulation, and study of complex networks. The package provides classes for graph objects, generators to create standard graphs, IO routines for reading in existing datasets, algorithms to analyze the resulting networks and some drawing tools.

The software takes as input two shape files: the first describes the topology, the seconds the source and sinks. The topology is either a mesh, representing the geography of the region or a polyline with just the roads net of the region. The roads are particularly important because aqueducts are built along roads for logistical reasons. The second file is a polygone file containing the buildings that should be served by the aqueduct and the water sources.

From those data, a first graph is obtained. The graph has as nodes the points described in topology file plus the buildings. The coordinates of bindings-representing nodes are the average of the coordinates of also have the metadata associated. The edges are the edges described in the topology file plus the edges connecting the building to the nearest node of the network in order to obtain a connected graph.

Bibliography

[1] Nauman Ahad, Junaid Qadir, and Nasir Ahsan. Neural networks in wireless networks: Techniques, applications and guidelines. *Journal of Network and Computer Applications*, 68:1 – 27, 2016.

[2] Mustafa K. Mehmet Ali and Faouzi Kamoun. Neural networks for shortest path computation and routing in computer networks. *IEEE transactions on neural networks*, 4 6:941–54, 1993.

[3] Dan Schult Aric Hagberg and Pieter Swart. *NetworkX Reference*. 2017.

[4] Julio Barbancho, Carlos León, F.J. Molina, and Antonio Barbancho. Using artificial intelligence in routing schemes for wireless networks. *Computer Communications*, 30(14):2802 – 2811, 2007. Network Coverage and Routing Schemes for Wireless Sensor Networks.

[5] Arson Chloé. Bio-inspired geomechanics. 2017.

[6] Vanessa M. Dunbabin, Johannes A. Postma, Andrea Schnepf, Loïc Pagès, Mathieu Javaux, Lianhai Wu, Daniel Leitner, Ying L. Chen, Zed Rengel, and Art J. Diggle. Modelling root–soil interactions using three–dimensional models of root growth, architecture and function. *Plant and Soil*, 372(1):93–124, Nov 2013.

[7] Lionel Dupuy, Peter J. Gregory, and A. Glyn Bengough. Root growth models: towards a new generation of continuous approaches. *Journal of Experimental Botany*, 61(8):2131–2143, 2010.

[8] Yung hsiang Chen and Fi-John Chang. Evolutionary artificial neural networks for hydrological systems forecasting. *Journal of Hydrology*, 367(1):125 – 137, 2009.

[9] R. Joobbani. *An artificial intelligence approach to VLSI routing*. Kluwer Academic Publishers,Norwell, MA, Jan 1985.

[10] Er.Manpreet Kaur. A review on neural network and ant colony optimization for vehicle traffic analysis and routing. 06 2017.

[11] Ben Kröse, Ben Krose, Patrick van der Smagt, and Patrick Smagt. An introduction to neural networks. 1993.

[12] Holger R. Maier and Graeme C. Dandy. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental Modelling Software*, 15(1):101 – 124, 2000.

[13] Holger R. Maier, Ashu Jain, Graeme C. Dandy, and K.P. Sudheer. Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions. *Environmental Modelling Software*, 25(8):891 – 909, 2010.

[14] Jacek Mandziuk. Solving the travelling salesman problem with a hopfield-type neural network. *Demonstratio Mathematica-Politechnika Warszawska*, 29:219–231, 1996.

[15] Muhammad Mustafa. Artificial neural networks modeling in water resources engineering: Infrastructure and applications. 62:341–349, 01 2011.

[16] Alain Pierret, Claude Doussan, Yvan Capowiez, Francois Bastardie, and Loïc Pagès. Root functional architecture: A framework for modeling the interplay between roots and soil. 6, 05 2007.

[17] Jean-Yves Potvin. State-of-the-art survey—the traveling salesman problem: A neural network perspective. *ORSA Journal on Computing*, 5(4):328–348, 1993.

[18] K.C. Tan, L.H. Lee, and K. Ou. Artificial intelligence heuristics in solving vehicle routing problems with time window constraints. *Engineering Applications of Artificial Intelligence*, 14(6):825 – 837, 2001.

[19] Atsushi Tero, Seiji Takagi, Tetsu Saigusa, Kentaro Ito, Daniel Bebber, Mark Fricker, Kenji Yumiki, Ryo Kobayashi, and Toshiyuki Nakagaki. Rules for biologically inspired adaptive network design. 327:439–42, 01 2010.

[20] Michal Turcanik. Network routing by artificial neural network. *2012 Military Communications and Information Systems Conference (MCC)*, pages 1–5, 2012.

[21] E. Wacholder, J. Han, and R. C. Mann. A neural network algorithm for the multiple traveling salesmen problem. *Biological Cybernetics*, 61(1):11–19, May 1989.

[22] Steven Willmott and Boi Faltings. Workshop note: Artificial intelligence network routing problems. 01 2018.

[21] [6] [7] [22] [15] [16] [10] [19] [17] [8] [4] [18] [1] [13] [12] [2] [11] [14] [20] [9] [5] [3] [**?**]