POLITECNICO DI MILANO
Corso di Laurea (MAGISTRALE ?) in Ingegneria Informatica
Dipartimento di Elettronica e Informazione

TITOLO DELLA TESI

AI & R Lab
Laboratorio di Intelligenza Artificiale
e Robotica del Politecnico di Milano

Relatore: Ing. Matteo Matteucci
Correlatore: Prof. Andrea Bonarini

Tesi di Laurea di:
Bud Spencer, matricola 634845
Terence Hill, matricola 641067

Anno Accademico 200N-200N+1

In remembrance of Jean-Baptiste...

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.

Acknowledgements

Ringrazio ....

Chapter 1

Introduction

L'introduzione deve essere atomica, quindi non deve contenere nè sottosezioni nè paragrafi nè altro. Il titolo, il sommario e l'introduzione devono sembrare delle scatole cinesi, nel senso che lette in quest'ordine devono progressivamente svelare informazioni sul contenuto per incatenare l'attenzione del lettore e indurlo a leggere l'opera fino in fondo. L'introduzione deve essere tripartita, non graficamente ma logicamente:

1.1 General context

La prima parte contiene una frase che spiega l'area generale dove si svolge il lavoro; una che spiega la sottoarea più specifica dove si svolge il lavoro e la terza, che dovrebbe cominciare con le seguenti parole "lo scopo della tesi è ...", illustra l'obbiettivo del lavoro. Poi vi devono essere una o due frasi che contengano una breve spiegazione di cosa e come è stato fatto, delle attività sperimentali, dei risultati ottenuti con una valutazione e degli sviluppi futuri. La prima parte deve essere circa una facciata e mezza o due

1.2 Motivation and Objectives

1.3 Brief description of the approach

La seconda parte deve essere una esplosione della prima e deve quindi mostrare in maniera più esplicita l'area dove si svolge il lavoro, le fonti bibliografiche più importanti su cui si fonda il lavoro in maniera sintetica (una pagina) evidenziando i lavori in letteratura che presentano attinenza con il lavoro affrontato in modo da mostrare da dove e perché è sorta la tematica di studio. Poi si mostrano esplicitamente le realizzazioni, le direttive future di ricerca, quali sono i problemi aperti e quali quelli affrontati e si ripete lo scopo della tesi. Questa parte deve essere piena (ma non grondante come la sezione due) di citazioni bibliografiche e deve essere lunga circa 4 facciate.

1.4 Structure of the reporto

La terza parte contiene la descrizione della struttura della tesi ed è organizzata nel modo seguente. "La tesi è strutturata nel modo seguente.

Nella sezione due si mostra ...

Nella sez. tre si illustra ...

Nella sez. quattro si descrive ...

Nelle conclusioni si riassumono gli scopi, le valutazioni di questi e le prospettive future ...

Nell'appendice A si riporta ... (Dopo ogni sezione o appendice ci vuole un punto)."

I titoli delle sezioni da 2 a M-1 sono indicativi, ma bisogna cercare di mantenere un significato equipollente nel caso si vogliano cambiare. Queste sezioni possono contenere eventuali sottosezioni.

Chapter 2

State of the art

Nella seconda sezione si riporta lo stato dell'arte del settore, un inquadramento dell'area di ricerca orientato a portare il lettore all'interno della problematica affrontata. Bisogna dimostrare di conoscere le cose fatte fino ad ora in questo campo e il perché si sia reso necessario lo svolgimento di questo lavoro. Questa sezione deve essere grondante di citazioni bibliografiche [**?**].

Chapter 3

Approach

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. 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. 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 $illuminationreducesphysarum'sgrowth$. 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. The use of Root Architecture Models was abandoned in order to investigate the use of Machine Learning, more specifically, Artificial Neural Networks.

ARTIFICIAL NEURAL NETWORKS

Copier la partie avec formules [TODO]

Artificial Neural Networks are not the best suited tool to solve routing problems. First of all, creating an ANN is a difficult task. Therefore, another method is worth considering. In addition, it is not possible to know whether the ANN will deliver a solution that will converge and whether this solution is optimal. Thus ANNs as a poorly efficient way to solve the problem.

Nonetheless, machine learning can still be useful. Indeed, the interpretation of geographical information files can be done through clustering algorithms.

Chapter 4

Progetto logico della soluzione del problema

4.1 Problem staterment and modelisation

The overall idea of our software solution is to connect water sources and consumers to a network of pipes in the most efficent way possible. Indeed the definition of the best net is a key problem. 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. We can now more formaly define our poblem. Being a topography a graph representing the meshed surface of a region, the problem of designing an acqueduct as the one of finding the recovering-graph on the topograpy graph connecting water consumers and sources. We will use the euclidian metric on a space with tree dimensions.

4.2 Data reading

In **??** is shown to the flow chart of our sofware. The imput is composed of three geografical files: mesh represent the topology of the studied region, source-sinks contains the locations for each water source and consumer (called sink from now on). The last file describes the roads network in the area as the pipes should preferebly run along roads for is cheper to place them. For more informetion about the geographical data format, please refer to 5.1 Reading those input two graphs can be initialized: the topograpy and the aqueduct describings graphs. The first is a graph where nodes represent positions and edges the possible transitions between them. Transitions on roads will be preferrable. The aqueduct graph is initialized with source and sinks as nodes and no edges. An insight of the data structure we used to represent graph is given in section 5.2

4.3 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. Thanks to the particular nature of our problem a semplificatin is possible: 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. To achive this need to recognize group of buildings such as villages or neighbourhoods. Those sets are called sink clusters. This aproach caries muliple advantages. From the computetional point of view reduce the dimension of the sets on wich we run the routing algorithms. On the other side, once the two layers are identified we can use different strategies to connect the nodes, as we will se in the next paragraph. Efficent implemetations of clusterings algoritms are provided in scikit-learn. Scikit-learn is a well-known machine learning library for Python and it features various classification, regression and clustering algorithms. After this operation sinks are labeled as part of they respective cluster. A more detailed explenation can be found at 5.3.

4.4 Routing

We now 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. This operation is broken down in two tasks. First, find all the paths connecting sinks, than chose the smalest recouvring graph, i.e. the smalest acqueduct satisfing the specifics. To find the path connecting the sinks an optimal aproach is used on the topograpy graph, in our case the dijstrak algorithm. The lenght of those path and the traversed nodes are saved as attributes of edges of the aqueduct graph. Please pay attention to the fact that edges on the acqueduct graph are paths on the topograpy graph. This operation creates a complete graph for each cluster. Note that the set of those graph is a partition of the aqueduct graph. The second stage consists in eliminting the redundant edges: to do this we run the kruskal algorithm and calculate the minimum spanning tree. For more informetion read section 5.4. An other aproache, favoured in classical aqueducts design would be to calculate a partialy connected graph where certain nodes are connected to exactly one other node whereas others are connected to two or more other nodes. This makes possible to have some redundancy without the expense and complexity required for a connection between every node in the network. At this first stage of the project this part is left for further investigation. Considering now the adduction system a very similar aproach can be used: the clusters should be interlinked and connected to a source. Each distribution network, as is a tree, has a root node. We can initialize the adductin graph with all the cluster's roots nodes and the water sources. Finally this graph is connected with the tecnique previeusly used.

Chapter 5

Technical aspects

5.1 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. 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.

5.2 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 polygon file containing the buildings that should be served by the aqueduct and the water sources. From these 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.

5.3 Clustering

5.4 Routing

Chapter 6

Presentation and validation of experimental results

Si mostra il progetto dal punto di vista sperimentale, le cose materialmente realizzate. In questa sezione si mostrano le attività sperimentali svolte, si illustra il funzionamento del sistema (a grandi linee) e si spiegano i risultati ottenuti con la loro valutazione critica. Bisogna introdurre dati sulla complessità degli algoritmi e valutare l'efficienza del sistema.

Chapter 7

Conclusions

7.1 Summary of Thesis Achievements

Si mostrano le prospettive future di ricerca nell'area dove si è svolto il lavoro. Talvolta questa sezione può essere l'ultima sottosezione della precedente. Nelle conclusioni si deve richiamare l'area, lo scopo della tesi, cosa è stato fatto,come si valuta quello che si è fatto e si enfatizzano le prospettive future per mostrare come andare avanti nell'area di studio.

7.2 Applications

7.3 Future wok

Appendix A

Documentazione del progetto logico

Documentazione del progetto logico dove si documenta il progetto logico del sistema e se è il caso si mostra la progettazione in grande del SW e dell'HW. Quest'appendice mostra l'architettura logica implementativa (nella Sezione 4 c'era la descrizione, qui ci vanno gli schemi a blocchi e i diagrammi).

Appendix B

Documentazione della programmazione

Documentazione della programmazione in piccolo dove si mostra la struttura ed eventualmente l'albero di Jackson.