Mobile Agent Migration Strategy using Itinerary Graph and Neural Network

Nripesh Chauhan
Dept. Of Computer Engineering
NIT Kurukshetra, Haryana, India
E-mail:nir.chauhan@gmail.com

Manjeet Singh
Dept. Of Computer Engineering
NIT Kurukshetra, Haryana, India
E-mail:er.manjeet106@gmail.com
www.nitkkr.ac.in

Abstract

Mobile agents (MA) needs to migrate to several host to accomplish its task .Migration strategy is responsible for planning out an optimal migration path ,which ensures mobile agent to complete its task correctly and efficiently at the minimum cost .This paper describe the strategy adopted by a mobile agent to complete the task assigned by the user. In this paper we propose an idea in which MA makes use of Dependent Variable Hopfield Neural Network to select an optimal route for migrating from one place to another place.

Keywords: Mobile Agent, Migration Strategy, Itinerary graph, Neural Network, DVHNN.

1. Introduction

Mobile Agents [1] are autonomous and proactive software entities which act on behalf on an owner and have the ability to migrate through a heterogeneous network of computer .A mobile agent[4] can also decide when to migrate an which node to access. After migration to the desired host the mobile agent resume the execution of previously broken off or awaiting task. On completing the task, the mobile agent returns the result to the client. Therefore the client need not be constantly connected to the server. This will not only save lots of unnecessary transmission load. But also helps in the application of mobile calculation. The mobile agent can single execute all task assigned by the user. It can meet and interact with other agent when necessary while still executing its task. Migration. The shortest path problem is concerned with finding the shortest path from a specified origin to a specified destination in a given network while minimizing the total cost associated with path. The shortest path problem is an archetypical combinational optimization problem having widespread application in a variety of settings.

For large scale and real time application such as traffic routing and path, MA migration the existing series algorithm may not be effective due to the limitation of sequential processing in computational time. Therefore, parallel solution methods are more desirable. The dynamical systems approach to solving optimization problems using artificial neural networks has been greatly attracted due to the massively parallel operations of the computing units and very faster convergence properties. In this paper we have proposed an idea to combine neural networks with the concept of mobile agent. We make use of discrete-time recurrent neural network for finding the shortest path between two host and this path is used by the MA for moving from one place to another for performing his task.

2. Related Work

Currently there are 3 migration strategies [2].

(A)Ideal Migration Strategy: it is also called static migration. In this approach, a explicit itinerary graph is created based on its initial itinerary, after creating itinerary graph load information of itinerary graph is fetched. The drawback of this approach is that the change of load will outdate the edge and it requires a centralize server.

(B)One step Migration: This approach finds an optimal solution step by step during migration. It considers only the load on adjacent vertex. The drawback of this approach is that it does not provide the global optimal solution.

(C)Learning Migration Strategy[2]: This approach tries to find out a globally optimal solution ,in this approach during each time ,agents records the software and hardware load information of a network. Next time, agent can use those travel experiences accumulated during previous migration. Each time, the old experiences will be updated by new information collected by agent. The drawback of this approach is that it does not always provide globally optimal solution.

3. Itinerary planning for mobile agent[3]

A Mobile agent visit more than one host in a network which is called agencies to fulfill its task. Thus it needs a travel plan when its starts its journey, a usually fixed plan is provided by its owner. The plan should be dynamically changes when mobile agent moves from one host to another. headings should be typeset in boldface with the first letter of important words capitalized. Figure 1, shows the major infrastructure components.

3.1. *Map Module*

It is responsible for collecting data to create a local information base which is called map. Every agency manages its local map. Because of the given quantity it is not possible to store complete information of the network of all agencies. Hence the map is split into a local area map and a map of remote areas.

3.2.Router Planner Module

The Route Planner Module is able to calculate a shortest path through a network. This module uses the map data ,especially the data on connection topologies and qualities.

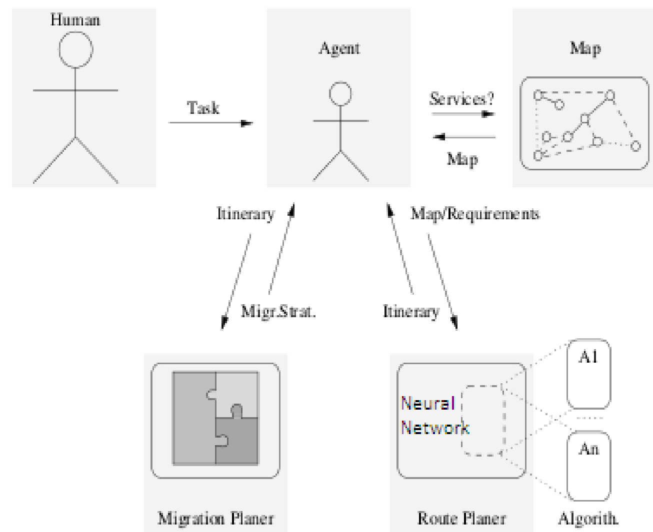

Figure1: Major Infrastructure Components

3.2. *Migration Planar Module*

It is designed to reallocate the itinerary to achieve an efficient journey. To resume execution of a migrated agent on a remote agency, the agent code with all referenced code parts is needed as well as the agents execution states and its current data.

4. Problem Statement and Dependent Variable Hopfield Neural Network:

Let G = (V, A) be a directed graph composed of a set of N number of vertices V and a set of M number of directed arcs A. A nonnegative number C_{rs} is associated with each arc (r, s) that stands for the cost from node r to node s. Non-existing arc costs are set to infinite. Often, for clarity of exposition, namely when referring to figures, we will label each existing arc with a unique index and denote its cost simply by C_i.

Let P_{sd} be a path from a source node s to a destination node d, defined as a set of consecutive nodes, connected by arcs in set A,

$$P_{sd} = \{s, n1, n2, \ldots\ldots, d\}$$

There are many applications of the direct graph since it can be used to model a wide variety of real-world problems. For example, in a road network, the vertices can represent intersections, the edges or arcs can represent streets, and the physical meaning of the cost can be the distance between the vertices. In this paper, the shortest path problem to be discussed is: find the shortest (least costly) possible directed path from a specified origin vertex to a specified destination vertex. The cost of the path is the sum of the cost coefficients on the edges in the path and the shortest path is the minimum cost path.
There is a cost associated with each path P_{sd} which consists of the sum of all partial arc costs participating in the path. The shortest path problem consists in finding the path connecting a given source-destination pair, (s,d), such that the cost associated with that path is minimum. The shortest path problem can be mathematically formulated as a linear integer program as follows [5].

$$\text{Minimize } \sum_{i=1}^{N} \sum_{j=1}^{N} Cij\ vij \qquad (1)$$

Subject to

$$\sum_{\substack{j=1 \text{ to } N \\ j \neq i \text{ and } (i,j) \text{exists}}} vij - \sum_{\substack{j=1 \text{ to } N \\ j \neq i \text{ and } (j,i) \text{exists}}} vji \qquad (2)$$

$$\text{And} \quad vij \in \{0,1\} \qquad (3)$$

Here, v_{ij} is the participation of the arc (i, j) in the path which can only be 0 or 1, i.e., the arc whether participates entirely or doesn't participate at all in the path. Non-existing arcs are not to be considered.

$$\emptyset i = \begin{cases} 1 & if\ i = s \\ -1 & if\ i = d \\ 0 & otherwise \end{cases} \qquad (4)$$

The set of N equations (2) can be stated in matrix form, though only $N - 1$ equations are linearly independent (thus, one of them is omitted). In addition, the double indexes in vij are replaced by corresponding single indexes variables. The resulting equation is then stated as:

$$\begin{bmatrix} a11 & a12 & \cdots & a1M \\ a21 & a22 & \cdots & a2m \\ \vdots & \vdots & \ddots & \vdots \\ aN-1,1 & aN-1,2 & \cdots & aN-1,M \end{bmatrix} \begin{bmatrix} v1 \\ v2 \\ \vdots \\ vM \end{bmatrix} = \begin{bmatrix} \emptyset 1 \\ \emptyset 2 \\ \vdots \\ \emptyset N-1 \end{bmatrix} \qquad (5)$$

$$Av = \emptyset;$$
$$vi \in \{0,1\}$$

In this equation, **A** is a $(N - 1) * M$ matrix which depends on graph configuration, $\emptyset$ is a vector with $(N - 1)$ elements, which enables path source and destination specification and **v** is a vector with M elements, where each one represents the participation of a single arc of the directed graph in the selected path.
The constraints just expressed in (5), that we will call Kirchhoff's constraints, are of considerable importance, because they are held by any valid solution to the SP problem (though the reciprocal is not necessarily true).
To build a NN that guarantees constraints (5), this equation is first rewritten using Gauss-Jordan elimination method.
The variables $v_1, \ldots\ldots, v_{N-1}$ are dependent variables and will be represented by dependent neurons in the NN to be presented next, while variables $v_N, \ldots\ldots, v_M$ are independent variables, that will be represented by independent neurons.
Consider as an example the graph represented in Figure 2: arcs (1, 3), (2, 4), (2, 5) and (3, 5) may be represented by independent neurons while neurons representing arcs (1, 2), (2, 3), (3, 4) and (3, 5) will then depend linearly on former neurons values, for this selection of dependent and independent variables.

It is well known that in order to design an recurrent neural network for optimization problems, one needs to construct an appropriate computational energy function so that the lowest energy state will correspond to the desired solutions. Based on the dual property of linear

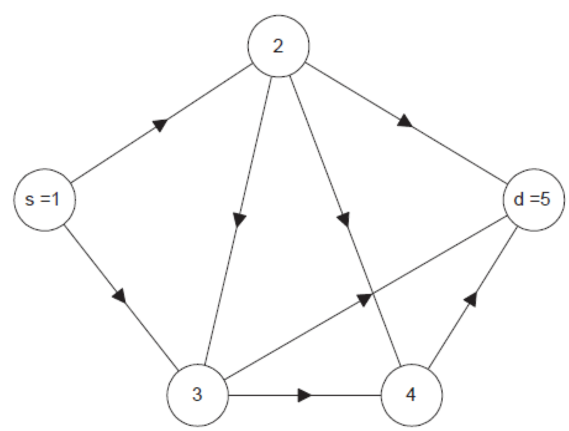

Figure 2: Graph Example

program[5], we consider an energy function for the dependent variable Hopfield neural network,

$$E = \sum_{1}^{M} \rho 1\, f(vi) \qquad (6)$$

The neuron model for the Dependent Variable Hopfield Neural Network is shown in figure 3:

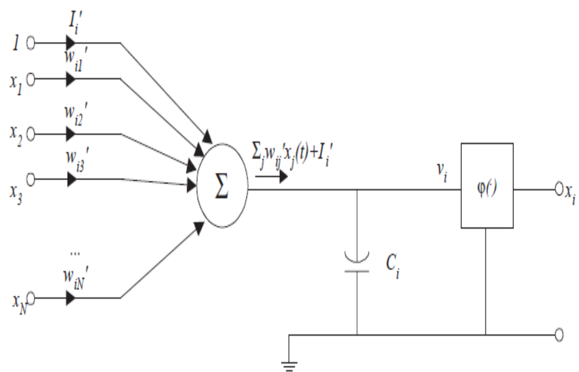

Figure 3: Neuron Model

The DVHNN for the graph given in figure (3) is given [6] in figure 4:
The Dependent Variable Hopfield Neural Network solves the shortest path problem with mixed-sign cost coefficients and is guaranteed to converge to an optimal solution globally [6].

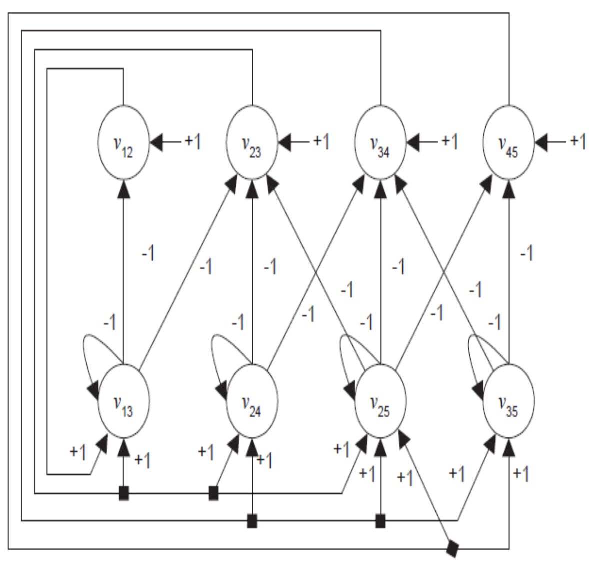

Figure 4: Example of Dependent Variable Hopfield Neural Network

5. Concluding Remarks

This paper introduces the concept of Neural Network in migration of Mobile agent. This paper also explains the Dependent Variable Hopfield Neural Network. This NN is proven to be capable of obtaining the shortest path routing for directed network with arbitrary cost coefficients, unlike some existing numerical algorithms. The shortest path calculated by NN is used by MA to migrate from one place to another place. Thus we can say that NN provides the optimal global solution of mobile agent migration problem.

References

1. Boyang, Da-youLiu, Kunyang, Sheng-sheng wang, "Strategically Migrating agents in itinerary graph", Procedding of the second international conference on machine learning and cybernetics ,xian,2-5 November 2003 .
2. You Jiang, Yiting jiang, You mingxia, "On the design and implementation of mobile agent migration strategy based on the extended itinerary graph", Information Science and service science (NISS),2011 5th international conference on new trends ,vol-2,pp 265-269.
3. Christian Erfurth, Wilhelm rossak, "Autonomous itinerary Planning for mobile agent", Friedrich-Schiller University Jena, 07/2003.
4. Garima Verma, "Strategies of mobile Agent for handling a task", International Journal of Engineering and innovative technology (IJEIT), 2012, vol-2(1).
5. M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, Linear Programming and Network Flows, 2nd ed. New York: Wiley, 1990.
6. Filipe Araujo, Bernardete Ribeiro, "A Neural Network for Shortest path Computation", Neural Networks IEEE Transaction volume 12, issue 5, pages 1067-1073, Sep 2001.