THE VEHICLE ROUTING PROBLEM : A NEURAL NETWORKS APPROACH

DAHBIA KADI
Groupe Ecole Supérieure de Commerce de Rennes, France
2, rue Robert d'Arbrissel 35000 Rennes
Tel : (02)99 54 63 71 Fax : (02) 99 33 08 24
Email : Dahbia.KADI@ESC-Rennes.fr
PhD student at INSA-Rennes, University of Rennes 1, France

LAOUCINE KERBACHE
Groupe Ecole Supérieure de Commerce de Rennes, France
2, rue Robert d'Arbrissel 35000 Rennes
Tel : (02)99 54 63 73 Fax : (02) 99 33 08 24
Email : Laoucine.KERBACHE@ESC-Rennes.fr
Visiting Professor at the University of Massachusetts, AMHERST, USA

Abstract

The Vehicle Routing Problem (VRP) is a complex combinatorial problem which has received great interest from the research community. Numerous exact algorithms have been developed to solve this problem but only for small size problems. For medium and large size VRPs, the use of heuristics is more appropriate. Neural Networks (NN) are among the most recent heuristics applied to the VRP. First, this paper examines the main classical exact algorithms and heuristics developed for the VRP. Second, different neural network approaches are presented with their application to the VRP and to the simpler case of the Travelling Salesman Problem. Finally, some research directions are pointed out and in this context, we present some of our preliminary results at this conference.

INTRODUCTION

The Vehicle Routing problem (VRP) plays an important role in the distribution management. Several real-life variants of the VRP exist such as the Vehicle Routing Problem with Time Windows and the Pickup and Delivery problem. The VRP shows a good interplay between theory and practice. Its complexity attracted the Operational Research community and has led to the development of numerous new algorithms. Small size problems are solved to optimality using exact methods. Unfortunately problems often encountered in the real world are larger hence intractable by these algorithms. That is why heuristics gain an important place in solving the VRP. Often they are the only alternative even though their solutions are sub-optimal.

Neural Networks are among the recent heuristics applied to the VRP. They are inspired from the biological functioning of the brain. Numerous papers consider their applications to optimisation problems but only a few concerns the Vehicle Routing Problem. The Travelling Salesman Problem (TSP) and data clustering are the closest optimisation problems to the VRP and that are solved by neural networks techniques. Hence, the understanding of neural networks developments in these two fields are necessary to extend them to the more complex VRP.

First, this paper examines the well known traditional algorithms developed for the solution of the Vehicle Routing Problem. These include the exact methods which lead to optimal solutions and the approximate methods which produce sometimes good feasible solutions for large size problems. Second, different neural networks approaches are presented with their application to the Vehicle Routing Problem and to the simplest Travelling Salesman Problem.

Finally, some research directions are pointed out. Some of these are presently investigated and preliminary results will be presented at the conference.

THE VEHICLE ROUTING PROBLEM

The vehicle routing problem has long been studied by operational research specialists. It is a fascinating problem which is easy to describe, but difficult to solve. It belongs to the class of NP-hard problems (Lenstra and Rinnoy Kan, 1981). The VRP can be defined as the problem of designing optimal delivery or collection of routes from one or several depots to a number of geographically dispersed cities or customers, subject to side constraints.

This problem can be considered as a more specialised version of the well-known Travelling Salesman Problem. The problem of the travelling salesman is to visit a number of towns (customers) in the shortest possible time or distance (Russel, 1977).

More generally, the ***generic vehicle routing problem*** involves the design of a set of minimum cost vehicle routes for a fleet of vehicles of known capacity which services a set of customers with known demands. All routes must originate and terminate at a common depot. Each customer is serviced exactly once and, in addition, all customers must be assigned to vehicles so that the total demand on any route does not exceed the capacity of the vehicle assigned to that route. Bodin *et al.* (1983) and more recently Laporte (1992a) and Osman (1993a) reviewed the problem through comprehensive surveys.

SOLUTION APPROACHES TO THE VEHICLE ROUTING PROBLEM

Several attempts have been used to solve the complex Vehicle Routing Problem. At first, researchers used exact methods but, facing the problem of the large size problems which remain unsolved by this approach, they quickly understood that the solution demands other flexible methods : heuristics. Even if these do not provide optimal solutions, their results are often feasible and sometimes very close to optimality. Presently, these approaches are the most widely used ones.

Exact methods

Exact algorithms, in the absence of round-off errors, will yield an exact solution in a finite number of steps. In general, all exact algorithms that solve the Vehicle Routing Problem are restricted by computer time and / or storage. Laporte and Nobert (1987), in a comprehensive survey, classified the exact methods for the VRP in three categories (i) direct tree search methods (ii) dynamic programming, and (iii) integer linear programming.

Direct tree search methods

Direct tree search methods consist of exploring the network or the tree formed by the customers (nodes) and the arcs (links). To use this approach the VRP is relaxed in a m-TSP or a k-degree centre tree. A detailed formulation of the m-TSP relaxation is given in (Laporte, 1992b). Problems involving up to 260 vertices were solved to optimality using this approach. The k-degree centre tree relaxation algorithm was developed by Christofides *et al.*, (1981) for symmetrical VRPs with m (fixed) vehicles. It consists of the construction of the spanning tree and then its formulation as an integer program representing the spanning tree and its constraints.

Dynamic programming

Dynamic programming is a traditional solution approach for constrained shortest path problems. This approach was proposed to solve the VRP by Eilon *et al.* (1971). The number of computations needed in this method is excessive in most cases due to the large number of states and transitions. The problems with more than 10 customers remain intractable for this approach. Christofides *et al.*, (1981a) used **State-space-relaxation** to reduce the number of states. This approach can be used with the capacitated VRP with up to 50 vertices (Christofides, 1985). More recently Dumas *et al.* (1995) showed the good performance of dynamic programming to optimally solve the travelling salesman problem with time windows for problems with up to 200 nodes.

Integer programming (Branch and Bound procedures)

The VRP can be formulated as a set partitioning problem or as a flow problem. The Set partitioning formulation was first proposed by Balinski and Quandt (1964). The formulation of this approach is an integer program (Houari, 1991). To overcome the difficulties inherent to an integer program, a **column generation algorithm** has been used by several researchers (Agarwal *et al.*, 1989 ; Haouri, 1991). Desrochers *et al.* (1992) proved that the algorithm was successful on a variety of practical sized benchmark VRP with Time Windows (VRPTW) test problems. The algorithm was capable of optimally solving 100-customer problems.
Several flow problem formulations are proposed for the VRP and the VRP with or without capacity and time constraints. Fisher and Jaikumar (1981) have developed a three-index vehicle flow formulation for the VRP with no identical vehicles, capacity and time windows restrictions. The complete formulation is given by Haouri (1991) and by Laporte (1992a).
A two-index flow formulation for the VRP can be derived from the three-index flow formulation by dropping the index that indicates which vehicle is used on a given route (Laporte *et al.* 1985).

Heuristics

Given the intrinsic difficulty of the VRP and the NP-hard class problems, in general and due to the limited success of the exact methods, approximate algorithms seem to offer the best alternative. There exists two kinds of heuristics applied to the VRP : The composite heuristics and the global heuristics.

Composite heuristics : route construction / local search improvement

In these heuristics, the solution is found by using first a construction heuristic that builds routes through the insertion of new customers and then a local route improvement heuristic that modifies the location of the customers within existing routes through exchange procedures.
The constructive methods are used to find an initial feasible solution to the routing problem. The construction of routes can be sequential or parallel. The sequential method constructs the routes one by one. The most known ones are : the insertion method and the nearest neighbour method (Desrochers *et al.*, 1988 ; Solomon, 1987).
The parallel methods usually start with a number of routes and progressively connect some of them like the savings algorithm (Clarke and Wright, 1963 ; Dantzig and Ramser, 1959). Solomon (1987) applied this approach to the VRPTW.
Tour improvement methods were first applied to the TSP. The improvement is in general made by an exchange of customers or links in the same route or between different routes (Christofides and Eilon, 1969). Because of the prohibitive computational time needed when many exchanges are considered, an 2 or 3 exchange or an Or-exchange (Or, 1976) are mostly used.
The two-phase algorithm is a method which uses two stages to solve the VRP : a clustering stage and a scheduling stage. Two methods have been used : route-first cluster-second and cluster-first route-second (Gillett and Miller, 1974 ; Christofides *et al.*, 1979 ; Fisher and Jaikmur, 1981 ; Toth, 1984 ; Solomon, 1987).

Global heuristics

Global heuristics combine both route construction and route improvement in the same phase.

a) Metaheuristics Tabu search and simulated annealing

Tabu search and simulated annealing are search schemes in which successive neighbours of a solution are examined, and the objective is allowed to deteriorate in order to avoid local minima. (Osman, 1993b).
Tabu Search (TS) was first proposed by Glover (1977). Gendreau *et al.* (1994) developed an efficient tabu search algorithm for VRP with capacity and route length restrictions. This heuristic outperforms the best existing heuristics and often produces the best known solutions. Recently, Carlton and Barnes (1996) applied a robust Tabu Search heuristic to the TSP with Time Windows.
Simulated annealing (SA) algorithm was first used by Kirkpatrick *et al.* (1983) in the combinatorial optimisation. To avoid the problem of the local minima of the descent methods, SA accepts non improvement moves with certain probabilities. These probabilities are determined by a control parameter T called a temperature (Osman, 1993b).
The major problem with these two metaheuristics is the quite large running time which is not a polynomial function of the size of the input data.

b) Genetic Algorithms (GA)

Genetic algorithms have their origin in the observations of the genetic process in biology. To apply the genetic approach to a given problem, solutions to this problem must first be encoded as

chromosomes (bit strings). Then an evaluation function that returns the fitness or quality of each chromosome is defined. In general, the algorithm selects some parents chromosomes and generates offspring by exchanging bit strings to form a new generation. The population of chromosomes is normally improved by repeated replacement of the parents by better offspring which are selected using a fitness evaluation function.
Potvin *et al.* used the genetic algorithm as an improvement heuristic to identify an ordering of the customers that produces good routes (Potvin *et al.*, 1993 and 1996).

APPLICATION OF THE NN APPROACH TO SOLVE THE TSP AND THE VRP

Neural networks are mathematical models inspired by the functioning of biological neurones. Several types of the NN architecture have been developed. Among these the supervised learning algorithm has been extensively studied, namely through the back propagation algorithm (Rumelhart *et al.*, 1986) and Boltzman machine (Hinton *et al.*, 1984). Another architecture of NN is the unsupervised learning algorithms such as the self-organising networks among which the Kohonen networks are the most frequently used.
Many different classes of neural networks models are described in the literature. In the following sections we will present the three different models of neural networks : Hopfield NN, Multi-layers NN and Kohonen NN. The most important applications of those models to the Vehicle Routing Problem will also be presented. As there is very little work on the application of the NN to the VRP, most of the ideas used in the VRP are inspired from the application of the neural networks to the TSP. The improvement and the extension of these ideas are the basis of our work.

Hopfield Neural Networks applied to the TSP / VRP

Hopfield and Tank stated that problems that can be formulated in terms of desired optima, often subject to constraints, can be solved by a connected network of neurones, namely the Hopfield neural network (HNN). It is a fully connected recurrent network which consists of just one layer. Each neurone is input and output of this network and there is no hidden neurone.
Hopfield and Tank considered the TSP to illustrate the computational power of this network (Hopfield and Tank, 1985). To solve this problem they defined a network where each neurone represents a city in a particular position in a tour and the different connections represent the constraints and then determined an energy function to be minimised. This energy function is a ***lyapunov function*** of the system. In other words, the energy of the network decreases or remains the same at every iteration ensuring the progress toward a minimum, thus, it seems very natural to employ Hopfield Networks to the optimisation problems.
Hopfield and Tank reported having solved 10 and 30 cities problems but several difficulties appeared when other researchers tried to use their algorithm (Wilson and Pawly, 1988).
Later refinements to the Hopfield model greatly improved its performance for medium-sized TSPs.
The new refined approaches of the HNN guarantee feasible solutions and prevent being trapped in local minima (Aart and Korst, 1987 ; Cuykendall and Reese, 1989 ; Xu and Tsai, 1991).
Smith *et al* (1996) used a hybrid NN combining a self-organising NN and a Hopfield NN which eliminates infeasible solutions and solves non Euclidean problems. Their approach was applied to the car sequencing problem from the manufacturing industry (Smith *et al.*, 1996a and 1996b).

Multi-layers neural networks applied to the VRP

A multi-layer network consists of one input layer, one output layer and one or more hidden layers. In such a network the neurones belonging to the same layer are not connected to each other and the output of a neurone of the layer j is connected to a neurone of the layer i if $j<i$.

To respond to a stimulus, the signal is propagated from the input layer to the output layer by traversing the hidden layer. The response obtained by the output neurones is compared to the desired (theoretical) response and the error is then calculated. This error will serve to modify and correct the connection weights with the hidden layer and then the connection weights with the input layer (Levine, 1991).
Yu (1996) modelled the dispatching problem as a multiattribute choice problem and used a back-propagation neural network which, after a training phase on previous decisions taken by a human dispatcher, solves the problem of a courier service company operating in an urban area. The back-propagation Neural network learns a set of weights that approximately reproduces the dispatcher decision procedure. This work does not find the initial routes. It only studies the impact of changing the existing routes because of a lateness or a detour request.

The Kohonen Self-Organising Feature Maps (KSOM) applied to the VRP

The unsupervised learning allows a neural network to extract useful information only from the redundancy of the patterns which are presented to it. In contrast to the supervised neural network, the unsupervised one does not require explicit tutoring. It spontaneously self-organises upon presentation of input patterns. The property of this process is to build a mapping from a topological ordered set into another set preserving the topological proprieties of the first set. The Kohonen self organising maps are the most famous example of this kind of networks.
The Kohonen maps network is inspired from both competitive learning concepts and the modelling of the mammal perception systems in biology such as sight and hearing. Since 1984, Kohonen exploited two important observations from the hearing system to define the self-organising feature maps. He proposed a network architecture which takes in consideration the external signals and the internal connections of the neurones network.

The propagation activation rule of the Kohonen model is analogous to that observed in biology. Every biologic neurone of a sensory area interacts with neighbouring cells with a strength that depends on the distance from the considered cell. This dependence can be represented by the « Mexican hat » function.
As the weights of the internal connections of the network are represented by the « Mexican hat » the learning rule to define is that of the weights modification. Kohonen proposed a two-step rule :
- Election of the neurone that will correspond to a given input pattern.
- Augmentation of the activation of the elected neurone and its neighbourhood when the input signal is presented in order to make them closer to the input pattern.

Globally, the learning consists of giving similar output patterns to similar input patterns. As this learning is unsupervised, no desired output is supplied.

Solution of the VRP using only the KSOM neural networks does not exist, but it has already been solved by partially using this approach. In fact, one possible solution of the VRP consists of two steps, a general clustering problem of customers and a determination of the minimum cost of each route by re-organising customers who are in the same group. As these two problems have already been solved by the KSOM NN, the following two sections will examine first the problem of determining the shortest path which has received an extensive interest by researchers applying KSOM to optimisation problems and second the existing neural networks solutions to data clustering. From these works, we will draw possible extensions and future developments.

Shortest path problem and the Kohonen neural network

Several variations of the basic idea derived from Kohonen's Self Organisation principles and applied to the TSP exist. The most known ones are presented in the following.

a) Angéniol et al. KSOM Neural Network for the TSP (1988)

In the Angéniol *et al.* algorithm, a tour is given by a set of nodes joined together in a one dimensional ring. Their method differs from the traditional discrete algorithm which finds the solution by examining all the possible permutations of the cities. In the Angéniol *et al.* approach, all nodes are moving freely in the plane through an iterative process. A tour is obtained when every city has caught one node of the ring.

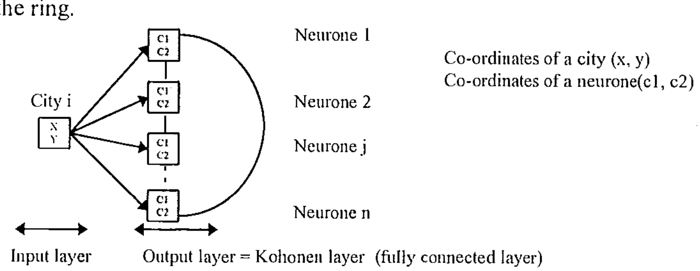

Figure 1 - The Angéniol et al. KSOM for the TSP

The Angéniol *et al.* algorithm proceeds by the presentation of the cities one by one. The node closest to the presented city moves towards it and induces its neighbours on the ring to do so as well, but with a decreasing intensity along the ring. This correlation between the motion of the neighbouring nodes intuitively leads to a minimisation of the distance between two neighbours, hence giving a short tour.

Results obtained by the authors were very good. They performed simulations on the same small sets of cities (10 and 30 cities) taken from Tank and Hopfield (1985) and from Durbin and Willshaw (1987). They obtained a good solution (less than 3% over the optimum) in two seconds using classical hardware. Their approach performed better than those of the elastic net method and other algorithms. The proposed algorithm scales well with problem size. An 1000-cities problem was solved on a sequential hardware in 20 minutes (Angéniol et al, 1988).

b) The Fort KSOM Neural Network (1988)

Fort applied the Kohonen algorithm to the travelling salesman problem. This algorithm considers a one dimensional circular array and puts it on the cities in such a way that two neighbour points of the array are also neighbour in distance. Those points are circular neighbours. He used the distances between a formal neurone i and a city j (Dij) and between a city j and a neighbour city k (Δjk) to measure the strength of the connection from a neurone i to a city j.

The neighbourhood of the winner node decreases with time allowing a faster convergence. Fort took the number of units N = 2n for small n (n=10 to 30) and N=2.5n for large n (n=200 to 400). This same idea was used in Durbin and Willshaw (1987).

The results obtained by his algorithm were worse than those of the Durbin and Willshaw elastic net algorithm and those of simulated annealing. A solution for (N+k) cities problem was obtained from a solution of a N-cities problem. This ability and the facility of the parallel implementation of their algorithm were stated as being the major advantages of the algorithm.

c) The adaptive ring of Hueter (1988)

Hueter used another variant of the Kohonen network. The number of nodes used on the ring is exactly the same as the number of cities. He solved the 10 and 30 cities problems given by Hopfield and Tank (1985).

The algorithm finds a solution by adapting a ring of N cities in the unit square. In this method, a city is chosen at random and the closest node to that city is determined. The positions of the other nodes are then adjusted according to the Kohonen learning law.
Three modifications were proposed by the author to improve this latter algorithm : smearing, attention and reallocation.
The smearing determines the probabilities of the nodes to win the competition for a selected city. Each node, at a distance r from that city, has the probability p(r) to win the competition.
The attention determines the chance of the node to win a competition. It increases the basin of a city's attraction for non-winning nodes. This brings nodes in a region of several cities and increases their win frequency to that of the norm. The attention technique is used when nodes are not far from each other. In contrast, when the most and the least active nodes are widely separated, a reallocation technique considered. The reallocation doubles the most active node and deletes the least active one from the net.
The results of this algorithm were compared to those of two methods. They are better than those of the Space Filling Curve but worse than those of simulated annealing. Although the author described the method, no comparison was made with the elastic net of Durbin and Willshaw.

d) The « conscientious » competitive learning net of DeSieno (1988)

DeSieno added a conscience to the competitive learning to improve the initial Kohonen learning. Based on the two drawbacks of the Kohonen learning, DeSieno proposed to add a conscience to overcome these problems. First, the basic Kohonen learning algorithm is biased in favour of the regions of lower density of input vectors. Second, the number of iterations required to reach a solution is very large because the losers slowly adjust their weight.
The conscience mechanism brings quickly all the elements available into the solution, and biases the competition process so that each element can win the competition with close to a desired probability of 1/N for an optimal vector quantization. This mechanism consists of the generation of the outputs of the competitive layer and the definition of a procedure for adjusting the weight vector. The winning element is the closest element to the presented input vector. This winner is not necessarily the element to have its weights reinforced. A bias is developed for each element based on the number of times it has won the competition.
In the beginning of the training, the algorithm acts as the Kohonen learning rule. When an element begins to win too often the DieSieno's algorithm rule is more effective.
The simulation results showed that the conscience learning law scales up linearly. This is an advantage for the large real-world problems. The training time required by the conscience mechanism for reproducing the input was about 1/10 of the training time needed by a basic Kohonen net. The algorithm was not tested on a TSP problem but the results are very interesting. They proved that the mechanism conscience improves the running time. Several researchers exploited these results to improve the KSOM for the TSP.

e) Fatava and Walker Kohonen neural net (1991)

Fatava and Walker showed that a simple modification of the self-organising Kohonen algorithm is capable of computing rapidly approximate solutions to the TSP. They solved problems up to 10 000 cities. The generated routes are reported to be slightly longer than those produced by simulated annealing ; compute time is lower and scales less than quadratically with problem size.
The algorithm is taken from the original work of Kohonen with very little modification. The network used consists of two one-dimensional layers. An input layer with three neurones and an output layer with n neurones. Each city presented to the net is formed by two co-ordinates and a third normalisation component, computed so that all input vectors have the same Euclidean length and no two input vectors are collinear.

The simulations were run on a VAX 3600, the results for the 30-cities of Hopfield and Tank problems were better than those of Hopfield and Tank.
For a set of a 30-cities problems the results were about 5% longer than those generated using simulated annealing but the algorithm ran faster.
They tested problems of 100, 500, 1000, 5000 and 10 000 cities and compared the results to those of the simulated annealing. They stated that the number of iterations converges when the number of cities increases and found that the growth of the necessary number of iterations decreases with problem size. They also extended the algorithm to multidimensional cases of the travelling salesman problem. The quality of the results are similar to those of the 2-dimensional case.

f) Guilty net and Vigilant net

The guilty net (Burke and Damancy, 1992 ; Burke 1994) is an adaptive neural network using exactly the same number of nodes as the number of cities. The nodes are located on a ring. The duplication/deletion of nodes process is replaced by a « conscience mechanism » of DeSieno (1988). The conscience mechanism disfavours the nodes that have won too often the competition and favours the losers. This mechanism ensures that each city is claimed by one unique node.
When a city is presented, the winner node is determined using the conscience mechanism. For each node a bias « bias j » is calculated. The winner node (j) is the one which satisfies the minimum of a biased distance. This mechanism allows a node to claim several cities in the beginning of the algorithm (bias =0) but after some iterations the conscience mechanism takes effect. The bias of the winniest nodes increases and disfavours the winning of these nodes. The most losing nodes are then given a chance to win a competition. At the end of the process each node will claim a unique city.
The guilty net emulates the nearest neighbour type heuristic. Simulation results on a 10-cities problem confirmed the node separation by the guilty net. Examples with 100 and 1000-cities are tested. The results are drawn from sets of five problems for 10, 30, 50 and 100-cities. They are worse than those of the Simulated annealing and elastic net but better than those of the Space Filling Curve and the Hopfield Tank algorithm. The advantage of this method is its faster computation time.

The vigilant net (Burke 1996) is an extension of the Guilty net. It is based on the node separation process. The idea is that if a node wins a competition but does not satisfy a particular criterion (a vigilance parameter), it is reset and a previously loser node is allowed to win. Vigilance is used in tour evolution as follows. On a particular pass through the list of cities, a node that wins a competition has « its win factor » increased by one. If the new win factor exceeds the vigilance parameter (p) then the node is reset and the competition is restarted for the loser nodes within its neighbourhood. Laura Burke chose a strict vigilance parameter p=2 allowing a node to be selected by two cities and, hence more freedom in node competition. This level can be increased or decreased on the basis of the ring length.
The model is tested on sets of 50, 100 and 200 cities with uniform or non uniform distributed data on a sequential materiel (PC-486 25 MH). The simulation results on the tour length were slightly better than those of the Space Filling Curve (SFC) and slightly worse than those of the MNN (Multiple Nearest Neighbour). In terms of computational time, the vigilant net is faster than the MNN and slower than SFC. Since the algorithm is a route construction algorithm, the speed of the algorithm is favoured against its solution quality.

The Clustering problem and the Kohonen neural network

The few articles applying neural networks to the VRP encountered in the literature concern the application of the KSOM neural network type to the clustering of the customers (routes determination). The literature shows that the KSOM neural networks give good solutions in the

clustering of the data compared to the more standard hierarchical clustering methods (Mangiameli *et al.* 1996). Hence the KSOM neural network can be used as the first step to determine the routes.
These methods use the same idea as that of the general data clustering. To cluster data by a KSOM neural network, the input is submitted to the network which clusters it into groups. Each node or neurone of the output layer represents a particular group. When an input data is presented to the network, the most correlated node j with this input wins the competition. The input data is then added to the group represented by the winning node j and the network updates its weights vector W to incorporate this input data into the group.

a) Hao et al. KSOM applied to the automated guided vehicle (1996)

The KSOM for data clustering has been used by Hao *et al.* (1996) for the automated guided vehicle (AGV). The AGV is a particular VRP in which the vehicle carries just one unit at a time but performs several moves. The proposed neural network model performs dispatching and routing tasks. It determines which requests are to be assigned to which material-handling devices based on the total distance/time minimisation objective. The authors proposed two algorithms based on Kohonen's self organising feature maps. The first adopts the strategy of routing first-clustering-second while the second one does interactive clustering and routing.

b) Potvin et al. neural network for the VRP with time windows (1996)

Potvin *et al.* solved the VRP with time windows using a neural network for the clustering phase. This is a pure constructive method. It is a parallel insertion in which the number of seed customers must be known a priori. This number is determined by a heuristic developed by Solomon (1987).
The Potvin *et al.* competitive neural network was used to cluster the customers. The input layer consists of the co-ordinates of the cities and the output layer consists of the clusters. The activation of each output unit is related to the distance between its weight vector and the presented input. The activation level is higher when this distance is smaller. The closest output unit to the input unit is the winner of the competition, and the input presented (i) is then associated to the group of the output winner (cluster j). The weight vector of the output winner w_j is adjusted to move even closer to the current customer (i), while the other weight vectors stay at their current location. The authors encountered the problem of the output units which never win the competition. To overcome it, they used the winning frequency of each node to bias the distance between a presented input and a cluster output. This process prevents the nodes from winning the competition too often (same idea used by the conscience mechanism of DeSieno, 1988, and the vigilance parameter of Burke, 1996).

CONCLUSION AND RESEARCH DIRECTIONS

Despite numerous applications of neural networks to different management fields, their application to the VRP is a relatively recent solution approach. In contrast, the TSP which represents a common benchmark for testing the applicability and the quality of most newly proposed methods for optimisation problems has been studied more often by researchers.
The application of the Hopfield neural network to the TSP proved to be useful but still limited because it is too expensive in time and storage. Medium and large problems remain unsolved by this approach even when parallel computing is used. To our knowledge, there is still no application of the Hopfield neural network to the VRP. The difficulty in using this approach is linked to, first, the definition of an energy function to describe the behaviour of the network and, second, the determination of a good HNN architecture to represent the VRP.
In contrast with the HNN, the application of the Kohonen Self Organising Maps neural network type to the TSP and the data clustering problems gives good results and looks promising for the future.

Several options already exist for improving the KSOM for the TSP and the clustering problem that can be exploited and extended to the VRP. Indeed, even if this approach is not appropriate for all the optimisation problems, it works very well with the Euclidean optimisation and the classification problems that form the VRP problem. Hence algorithms developed for the TSP and the clustering problem can be extended to handle the VRP. Solving the VRP using neural networks techniques developed for the TSP and the clustering problem is very promising. For instance, the good results obtained for the first two problems and the several improvements proposed to the initial KSOM to overcome some encountered drawbacks are very encouraging.
On one hand, the KSOM for clustering data has given good results compared to the classical clustering methods(Mangiameli *et al.*, 1996). On the other hand, the KSOM for solving the TSP showed that integrating some further strategies to the initial Kohonen algorithm like the Angéniol *et al.* process of duplication and deletion of nodes, the DeSieno conscience mechanism and its variants leads respectively to shortest path and reduced computing time.
This approach assumes that the VRP will be solved by, first, clustering the clients with respect to some constraints like the capacity of the vehicles and, then, finding the shortest path for each defined cluster. This reflects the parallel insertion process of the VRP.

Another possible development is the design of an hybrid KSOM and heuristic improvement approach. This idea has already been experimented by Potvin *et al.* (1996) who used both a neural network approach to cluster the data and genetic algorithms to improve the initial obtained solution. The use of a no-neural network improvement heuristic such as Simulated Annealing and Tabu Search is somewhat justified when the objective is to obtain a very good quality of the solution at the expense of computational time and storage.

Finally we propose the use of only one step KSOM to solve the VRP. The objective is to develop a good KSOM algorithm that determines not only a feasible but a good solution. The design of this algorithm requires efforts in two directions. First, the design of a modified KSOM architecture which permits a representation of the VRP and, second, the integration of some strategies in the neighbourhood function that leads to the overall shortest path and satisfy the VRP constraints at the same time.

REFERENCES

Agarwal, A., Mathur K. and Salkin H. (1989), A Set Partitioning Base Exact Algorithm for the Vehicle Routing Problem, **Networks**, 19, p. 731-749.

Angéniol, B., De La Croix Vaubois G. and Le Texier, J.Y. (1988), Self Organising Feature Maps and the Traveling Salesman Problem, **Neural Networks**, 1, p. 289-293.

Balinski, M.L and Quandt, R.E (1964), On an Integer Program for the Delivery Problem, **Operations Research**, Vol. 12, p. 300-304.

Bodin, L.D Golden, B.L Assad, A.A and Ball, M.O (1983), Routing and Scheduling of the Vehicles and Crews, **Compt. & Ops Res.**,Vol. 10, No. 2, pp 63-211, (Special Issue).

Burke, L. (1996), « Conscientious » Neural Nets for Tour Construction in The Travelling Salesman Problem : The Vigilant Net, **Compt. & Ops Res.**, Vol. 23, No. 2, p.121-129

Burke, L. (1994), Neural Methods for the Travelling Salesman Problem : Insights From Operations Research, **Neural Networks**, Vol. 7, pp.681-690.

Burke, L. and Damancy, P. (1992), The Guilty net for the Traveling Salesman Problem, **Computers Ops Res.** Vol. 19, No. 3/4, pp.255-265.

Carlton, W.B and Barnes, W. (1996), Solving the traveling-salesman problem with time windows using tabu search, **IE Transactions**, Vol 28, p. 617-629.

Christofides, N. and Eilon, S. (1969), An Algorithm for the Vehicle Dispatching Problem, **Operational Research Quarterly** 20, 309-318.

Christofides, N., Mingozzi, A. and Toth, P. (1979), The vehicle Routing Problem in **Combinatorial Optimization**, (N. CHRISTOFIDES, A. MINGOZZI, and P. Toth, C.Sandi), John Wiley, p.315-338, New York.

Christofides, N., Mingozzi, A.& Toth, P.(1981), Exact Algorithms for the Vehicle Routing Problem, Based On Spanning tree Shortest Path Relaxation, **Mathematical Programming** 20,255-282.

Christofides, N., Mingozzi, A., and Toth, P. (1981 a), Space State Relaxation Procedures for the Computation of Bounds to Routing Problems, **Networks** 11, 145-164.

Christofides, N. (1985), Vehicle Routing, in : **The Travelling Salesman Problem : A guided Tour of Combinatorial Optimization,** Lawer, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G. and Shmoys, A.H.G., Wiley.

Clarke, G., Wright, J.W. (1963), Scheduling Of Vehicles From A Central Depot To A Number Of Delivery Points, **Operations Research**, 12, p. 568-581.

Dantzig, G.B. and Ramser, J.H. (1959), The Truck Dispatching Problem, **Management Science**, p. 80-91.

DeSieno, D. (1988), Adding a Conscience to Competitive Learning, **Pro. Int. Conf. on Neural Networks I, 117-124, IEE Press**, New York.

Desrochers, M., Desrosiers, J. and Solomon, M. (1992), A New Optimization Algorithm For The Vehicle Routing Problem With Time Windows, **Operations Research**, Vol. 40, No. 2, p.342-354.

Desrochers, M. and Verhoog, T.W. (1991), A New Heuristic for The Feet Size and Mix Vehicle Routing Problem, Computers Operations Research, Vol. 18, No. 3, p. 263-274

Desrochers, M., Savelsbergh, M.W.P. and Soumis, F. (1988), Vehicle Routing With Time Windows: Optimization and Approximation, 1988, **EJOR**, 35, p. 242-254

Dumas, Y., Desrochers, J., Gelinas, E., Solomon, E. (1995), An Optimal Algorithm For The Traveling Salesman Problem With Time Windows, **Operations Research**, Vol. 43, No. 2.

Durbin, R. and Willshaw, D. (1987), An analogue approach to the traveling salesman using an elastic net method, **Nature**, 326, 689-691

Eilon, S., Watson-Candy, C.D.T. and Christofides, N. (1971) **Distribution Management, Mathematical Modeling and Practical Analysis**, Hafner New-York.

Fatava, F. and Walker, R. (1991), A Study of the application of Kohonen-type neural networks to the Travelling Salesman Problem, **Biological Cybernetics**, 64, 463-468.

Fisher, M.L. and Jaikumar, R. (1981), A Generalized Assignment Heuristic for Vehicle Routing, **Networks**, vol. 11, p. 109-124.

Fort, J.C. (1988), Solving a Combinatorial Problem Via Sel-Organizing Process : An Application of the Kohonen Algorithm to the Traveling Salesman Problem, **Biological Cybernetics**, 59, p.33-40.

Gendreau, M., Hertz, A. and Laporte, G. (1994), A Tabu Search Heuristic for the Vehicle Routing Problem, **Management Science**, Vol. 40, No. 10, p. 1276-1290.

Gillet, B.E and Miller, L.R. (1974), A Heuristic Algorithm For The Vehicle-Dispatch Problem, **Operations Research**, 22, p. 340-349.

Glover, F. (1977), Heuristics for Integer Programming Using Surrogate Constraints, **Decision Science**, Vol. 8, No. 1, p. 156-166.

Hao, G., Shang, J.S. and Vargas, L.G. (1996), A Neural Network Model for the Free-Ranging AGV Route Planning Problem, **Journal of Intelligent Manufacturing**, 7, p. 217-227.

Hinton, G., Sejnowski, T. and Ackeley, D. H. (1984), Boltzman Machine : Constraint Satisfaction Network That Learn, Technical Report CMU-CS-84-119, Department of Computer Science, **Carnegie-Mellon University.**

Houari, M. (1991), Les Problèmes de Tournées avec Fenêtre de temps, Modélisation et algorithmes de Résolution Exacte et Heuristique, Thèse de Doctorat, **Ecole Centrale des Arts et Manufactures, Paris.**

Hopfield, J.J. and Tank, D.W. (1985), Neural Computation of Decisions in Optimisation Problems, **Biological Cybernetics**, 5, 141-152.

Hueter, G.J.(1988), Solution of the Traveling Salesman Problem with Adaptive Ring, in Proceedings IEEE **International Conference on Neural Networks**, ICNN-88, San Diego, CA, p. 91-99.

Kirkpatrick, S., Gelatt, J.C.D. and Vecchi, M.P. (1983), Optimization by Simulated Annealing, **Science**, 220, 671-680.

Kohonen, T. (1988), An Introduction to Neural Computing, **Neural Networks**, 1, p.3-16.

Kohonen, T. (1984), Self-organisation and Associative Memory. Springer Series in Information Sciences. Spring-Verlag, New York.

Laporte, G., Nobert, Y. and Desrochers, M. (1985), Optimal Routing Under Capacity and Distance Restrictions, **Operational Research**, Vol. 33, p. 1050-1073.

Laporte, G., Nobert, Y. (1987), Exact Algorithms for the Vehicle Routing Problem, **Annals of Discrete Mathematics**, 31, p.147-184.

Laporte, G. (1992a), The Vehicle Routing Problem : An Overview of the Exact and Approximate Algorithms, **European Journal of Operational Research**, 59, p. 345-358.

Laporte, G. (1992b), The Travelling Salesman Problem : An overview Of Exact And Approximate Algorithms, **European Journal of Operational Research**, 59, 231-248.

Lenstra, J.K. and Rinnoy Kan, A.H.G. (1981), Complexity of Vehicle Routing and Scheduling Problem, **Networks**, Vol. 11, P 221-227.

Levine, D.S. (1991), **Introduction to Neural & Cognitive Modeling.**

Lin, S. and Kernighan, B.W. (1973), An Effective Heuristic Algorithm for the Traveling Salesman Problem, **Operations Research**, 21, pp. 498-516.

Mangiameli, P., Chen, S.K. and West, D. (1996), A comparison of SOM neural network and hierarchical clustering methods, **European Journal of Operational Research**, 93, 402-417.

Or, I. (1976), Travelling Salesman -Type Combinatorial Optimization Problems and their Relation to the Logistics of Regional Blood Banking, **PhD Dissertation, Northwestern University, Evanston**, Il, USA.

Osman, I. (1993a), Metastrategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problem, **Annals of Operations Research**, 41, p. 421-451.

Osman, I. (1993b), Vehicle Routing and Scheduling : Applications, Algorithms and Developments, **International Conference on Industrial Logistics, Rennes, France**.

Potvin, J.Y., Duhamel, C. and Guertin, F. (1993), A Genetic Algorithm for Vehicle Routing with Backhauling, **Université Blaise Pascal Clermont Ferrand II**, p.1-21.

Potvin, J.Y., Dube, D. and Robillard, C. (1996), A Hybrid Approach To Vehicle Routing Using Neural Networks And Genetic Algorithms, **Applied Intelligence**, 6, p.241-252.

Rumelhart, D.E., Hinton, G. and Williams, R. (1986), Learning Internal Representations By Back-Propagation Errors, **Nature**, 323, p. 533-536.

Russel, R.A. (1977), An Effective Heuristic for the M-Tour Traveling Salesman Problem with Some Side Conditions (Technical notes), **Operations Research**, Vol. 4, No. 6, p. 711-719.

Smith, K., Palaniswami, M. and Krishnamoorthy, M. (1996a), Traditional heuristic versus Hopfield neural network approaches to a car sequencing problem, **European Journal of Operational research**, 93, 300-316, 1996

Smith, K., Palaniswami, M. and Krishnamoorthy, M. (1996b), A Hybrid Approach to Combinatorial Optimisation, **Computers Ops. Res.** Vol. 23, No. 6, pp. 597-610.

Solomon, M.M. (1987), Algorithms for the Vehicle Routing Problems with Time Windows Constraints, **Operations Research**, Vol. 35, No. 2, p. 254-265.

Wilson, G.V. and Pawley, G.S. (1988), On the Stability of the Travelling Salesman Problem Algorithm of Hopfield and Tank, **Biological Cybernetics**, Vol. 58, p.63-70.

Xu, X. and Tsai, W.T. (1991), Effective Neural Algorithms for the Traveling Salesman Problem, **Neural Networks**, Vol. 4, pp. 193-205.

Yu, S. (1996), Neural Networks for the Vehicle Routing Problem, **Dissertation thesis, Dept of Computing and Operations research, University of Montreal**, Canada.