Neural Formulation of Functional Annealing and Application to Traveling Salesman Problem

Domingo López-Rodríguez and Enrique Mérida-Casermeiro

Department of Applied Mathematics
University of Málaga
Málaga, Spain
e-mail:{dlopez,merida}@ctima.uma.es

Abstract. A new technique based on slight modifications of the energy function of a neural network is proposed in this work. The aim of this technique is to reduce the number of local minima of the objective function, allowing the net to improve considerably the quality of the obtained solutions. Most discrete neural networks can benefit from its application, due to its generality and ease of use. We also propose its theoretical bases, and its application to recurrent neural networks is shown. To this end, Traveling Salesman Problem has been used as benchmark for this technique, as it is the best-known combinatorial optimization problem and the most used benchmark to quantify the efficiency of algorithms applied to this kind of problems. This new method has been compared to other techniques in terms of average solution quality, showing a substantial improvement. It is also able, in most of the performed simulations, to reach solutions near to global optimum.

1 Introduction

Traveling Salesman Problem (TSP) is one of the most well-known and studied combinatorial optimization problems due to its wide range of real-life applications and the intrinsic complexity it possesses.

Real-life applications cover aspects such that automatic routing for robots and hole location in printed circuits design [15], as well as gas turbine checking, machine task scheduling or crystallographic analysis [3], among others.

This problem can be stated as follows: given N cities $X_1, \ldots, X_N \in \mathbb{R}^2$ and distances $d_{i,j}$ between each pair of cities X_i and X_j, the objective is to find the shortest closed tour visiting each city once.

Despite the simplicity of this definition, this problem is one of the most typical representatives of the NP-complete complexity class, indicating its degree of difficulty in resolution. Thus, there is need of algorithms that reaches good approximations to the optimal solution with little time consumption.

For this reason, in addition to classical methods of Operations Research and Optimization, several different methods have been developed, including genetic algorithms [13], simulated annealing [1], tabu search[4], and neural networks [14].

Concerning neural networks, the main subject to deal with is getting an adequate representation or formulation of the problem such that its resolution arises as an energy function minimization problem.

Hopfield and Tank [5] in 1985 proposed the first neural network to deal with combinatorial optimization problems (the analog Hopfield model) which was used precisely to solve this problem, TSP.
This analog model has more ability to escape from local minima than the discrete model. Some deficiencies are present in both models, as the necessity to *a priori* fine-tune a high number of parameters in the energy function, as pointed out by Wilson and Pawley [17].
Other approaches are entirely based on Kohonen's self-organizing map [6], getting the best performance among all of them the so-called KNIES network [2], where some explicit statistics where incorporated to the original model. This network also presented the drawback of fine-tuning a number of parameters in order to achieve good results.
In the recent years, a new model of multivalued network has been presented by Mérida et al. [7], MREM, generalizing bipolar Hopfield and other multivalued models [9]. Results given by this network proved to improve those given by KNIES, and presenting the advantage of no fine-tuning of parameters, contrary to KNIES. This model has also been applied to other combinatorial optimization problems [10–12].
MREM computational dynamics is based in a heuristic technique that detects the possible crosses in the tour and try to undo them, minimizing the total distance of the tour.
The aim of this work is to present an additional technique for this model MREM to be able to escape from certain local minima, improving so its efficiency when dealing with problems presenting difficulties, like the one studied in this paper.

2 The multivalued MREM model

The MREM neural model consists of a series of multivalued neurons, where the state of i-th neuron is characterized by its output s_i, taking any value in a finite set, denoted by $\mathcal{M}$. This set of possible outputs does not need to be numerical. Both quantitative and qualitative elements may be included in $\mathcal{M}$, for example, 'true' and 'false' or 'red', 'green' and 'blue'.
Network state is completely determined by a state vector $\boldsymbol{S} = (s_1, s_2, \ldots, s_N) \in \mathcal{M}^N$, where N is the number of neurons in the net.
An energy function is associated to each state of the net, and is defined in the following terms:

$$E(\boldsymbol{S}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{i,j} f(s_i, s_j) \tag{1}$$

where $W = (w_{i,j})$ is a $N \times N$ matrix representing the connection between the different neurons ($w_{i,j}$ is the weight that neuron j makes on neuron i) and $f : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is a similarity function, that is, $f(s_i, s_j)$ represents a measure of the similarity between outputs of neurons i and j. Note that MREM with only two states ($\{-1, 1\}$ or $\{0, 1\}$) and $f(x, y) = xy$ is precisely the bipolar Hopfield model.
The purpose of the net is to minimize the energy function described before. To this end, a random initial state $\boldsymbol{S}_0$ is introduced into the net and at time t, state vector $\boldsymbol{S}(t)$ will be changed for another state vector $\boldsymbol{S}(t+1)$ (defined by the computational dynamics) if $E(\boldsymbol{S}(t+1)) < E(\boldsymbol{S}(t))$. If there is not any $\boldsymbol{S}'$

in the neighborhood of $\boldsymbol{S}(t)$ such that $E(\boldsymbol{S}') < E(\boldsymbol{S}(t))$, then $\boldsymbol{S}(t)$ is a local minimum (for the considered dynamics) and the net stops iterating.
Several dynamics (both sequential [7] and parallel [8]) are allowed in this model, due to its generality, although the most adequate dynamics will be determined by the problem to be tackled.

3 MREM for the Traveling Salesman Problem

In order to get the TSP solved by this neural model, two identifications must be done:

- **A network state must be identified to a solution to the TSP:**
  A solution to the TSP can be represented as a permutation in the set of numbers $\{1, \ldots, N\}$, where N is the number of cities. For this reason, the net will be formed by N neurons, each of them taking value in the set $\mathcal{M} = \{1, \ldots, N\}$, such that state vector $\boldsymbol{S} = (s_1, \ldots, s_N)$ represents a permutation of $\{1, \ldots, N\}$ (feasible state). With this representation, $s_i = k$ means that k-th city will be visited in the i-th place.
- **The energy function must be identified to the total distance of a tour:**
  On the other hand, if in Eq. (1) we let $f(x, y) = -2d_{x,y}$ and

$$w_{i,j} = \begin{cases} 1 & (j = i + 1) \vee ((i = N) \wedge (j = 1)) \\ 0 & \text{otherwise} \end{cases}$$

  the energy function obtained is $E(\boldsymbol{S}) = \sum_{i=1}^{N-1} d_{s_i, s_{i+1}} + d_{s_N, s_1}$, the total distance of the tour represented by state vector $\boldsymbol{S}$.

Computational dynamics is based on starting with a random feasible initial state vector and updating neurons outputs to keep the current state vector inside the feasible states set. To this end, at each iteration, a *2-opt* update will be made on current state vector, that is, every pair of neurons, p, q with $p > q + 1$, is studied and checked whether there exists a cross between segments $[s_p, s_{p+1}]$ and $[s_q, s_{q+1}]$. In this case, the next relation holds:

$$d_{s_p, s_{p+1}} + d_{s_q, s_{q+1}} < d_{s_p, s_q} + d_{s_{p+1}, s_{q+1}}$$

Then, the trajectory between cities s_{p+1} and s_q is inverted, that is, if $\boldsymbol{S}$ is the current state, the new state vector $\boldsymbol{S}'$ will be defined by:

$$s'_i = \begin{cases} s_{q+p+1-i} & \text{if } p + 1 \leq i \leq q \\ s_i & \text{otherwise} \end{cases}$$

As an additional technique for improvement, it has also been considered *3-opt* updates: the tour is decomposed into three consecutive arcs, A, B and C, which are then recombined in all possible ways: $\{ABC, ACB, A\hat{B}C, AB\hat{C}, A\hat{B}\hat{C}, A\hat{C}B, AC\hat{B}, A\hat{C}\hat{B}\}$, where $\hat{A}$, $\hat{B}$, $\hat{C}$ are the reversed arcs corresponding to A, B, and C, respectively. Note that $\{ABC, A\hat{B}C, AB\hat{C}, A\hat{C}\hat{B}\}$ are *2-opt* updates, so there is no need to check them again.
The next state of the net will be the combination that decreases most the energy function. For more details, see [7].

4 Functional Annealing

In this Section, despite its neural application, we will rigorously present this optimization method, Functional Annealing (FA), giving basic theorems and results guaranteeing its convergence, although not including all the proofs, due to the restriction in the length of this paper.

4.1 Theoretical Foundations

Suppose that a function $F : V \to \mathbb{R}$ is to be minimized, where V is a discrete set (not necessarily numerical). We will study the possibility of some slight modifications to this function F in order to make easier its minimization.
So, a sequence $\{F_n\}_{n\geq 1}$ of functions defined over the same set V is considered. In addition, these two conditions will also be assumed in the rest of this work:

1. For each $x \in V$, $F_n(x) \to F(x)$.
2. $\sum_{n=1}^{\infty} ||F_n - F_{n+1}||_\infty < \infty$

where $||g||_\infty = \max_{x\in V} |g(x)|$.
We will assume that an iterative optimization technique is used to minimize each of the functions F_n, starting from a random point $x_1^{(n)}$ and creating a sequence $x_k^{(n)}$ such that que $F_n(x_k^{(n)}) \geq F_n(x_{k+1}^{(n)})$, that verifies $\lim_{k\to\infty} x_k^{(n)} = x_*^{(n)}$.
By using this way of building sequence $x_k^{(n)}$, convergence of FA is assured by the next result:

Theorem 1. *Under the above assumptions, if we let* $x_1^{(n+1)} = x_*^{(n)}$, *with a random* $x_1^{(1)}$, *then the sequence* $F_n(x_*^{(n)})$ *is convergent.*

This result states that if $x_*^{(n)}$ is taken as initial guess to minimize function F_{n+1}, that is, $x_1^{(n+1)} = x_*^{(n)}$, then the value of the successive objective functions evaluated at $x_*^{(n)}$ converges. Its limit is also a value of the original objective function F.

Corollary 1. *There exists* $x_* \in V$ *such that*

$$F(x_*) = \lim_{n\to\infty} F_n(x_*^{(n)}) = \lim_{n\to\infty} F(x_*^{(n)})$$

A trivial result states that if $x_*^{(n)}$ is global minimum of F_n for all $n \geq n_0 \in \mathbb{N}$, then x_* is global minimum of F.
This global result is hard to be verified. Some less restricting results are presented next:

Lemma 1. *Given* $x, x' \in V$ *with* $F(x) < F(x')$, *there exists* n_0 *such that if* $n \geq n_0$ *then* $F_n(x) < F_n(x')$.

Corollary 2. *There exists* $n_0 \in \mathbb{N}$ *such that for all* $x, x' \in V$ *with* $F(x) < F(x')$, *it is obtained* $F_n(x) < F_n(x')$ *for all* $n \geq n_0$.

Proposition 1. *There exists* $n_0 \in \mathbb{N}$ *such that if* $n \geq n_0$, *then* $F_n(x) \leq F_n(x')$ *implies that* $F(x) \leq F(x')$.

Proof. Let $\varepsilon = \min\{F(x) - F(x') : x, x' \in V \text{ and } F(x) - F(x') > 0\} > 0$. Since V is a discrete (finite) set, ε is well-defined. This definition implies that if x, x' are such that $F(x) - F(x') < \varepsilon$, then $F(x) - F(x') \leq 0$.
Now, let $0 < \varepsilon' < \varepsilon$. Since $F_n(x) \to F(x)$ for all x, given x, x', there exists a natural number $n_{(x,x')}$ (depending on x, x') such that if $n \geq n_{(x,x')}$, then $|F(x) - F_n(x)| < \frac{\varepsilon'}{2}$ and $|F(x') - F_n(x')| < \frac{\varepsilon'}{2}$
Particularly, $[F(x) - F(x')] - [F_n(x) - F_n(x')] < \varepsilon'$, so $F(x) - F(x') < \varepsilon' + F_n(x) - F_n(x')$.
Suppose $F_n(x) \leq F_n(x')$, that is, $F_n(x) - F_n(x') \leq 0$, then, by substituting in the expression above, we arrive at $F(x) - F(x') \leq \varepsilon' < \varepsilon$, and, as explained at the beginning of the proof, we have $F(x) - F(x') \leq 0$, that is, $F(x) \leq F(x')$.
So, by taking $n_0 = \max\{n_{(x,x')} : x, x' \in V\}$, the result is obtained.

For this reason, it will not be necessary (in practice) to build the whole sequence F_n, but it will suffice to minimize until certain degree of approximation indicated by F_{n_0}. In addition, this last result implies the following one:

Corollary 3. *If $x_*^{(n)}$ is a local minimum of F_n for all n greater than a certain $n_0 \in \mathbb{N}$, then x_* is a local minimum of F.*

It must be noted that every convergence result above only states the convergence of $\{F_n(x_*^{(n)})\}$, to reach the conclusion of $F(x_*)$ being a local minimum of F. But, under these so general assumptions, it is not strictly true that $x_*^{(n)} \to x_*$. Let us give an example:

Example 1. Let $M \in \mathbb{N}$ and $F : \{0, \frac{1}{M}, \ldots, \frac{M-1}{M}, 1\} \to \mathbb{R}$ be the function given by $F(x) = 0$ for every $x \in V$.
Sequence F_n will be defined in the following terms:

- If n is odd, F_n will be the function whose graph is the segment of extremes $(0, \frac{1}{n})$ and $(1, \frac{1}{n+1})$, evaluated at points in V. Its analytical expression is: $F_n(x) = \frac{1}{n} - \frac{1}{n(n+1)}x$. It is convex and its global minimum is reached at $x_*^{(n)} = 1$.
- If n is even, F_n will be the function whose graph is the segment with extremes $(0, \frac{1}{n+1})$ and $(1, \frac{1}{n})$, evaluated in V. Its analytical expression is: $F_n(x) = \frac{1}{n+1} + \frac{1}{n(n+1)}x$. It is convex and its global minimum is reached at $x_*^{(n)} = 0$.

Obviously, $F_n(x) \to F(x)$ for all $x \in V$. Let us verify that the assumption $\sum_{n=1}^{\infty} ||F_n - F_{n+1}||_\infty < \infty$ holds.
It can be easily proved that $||F_n - F_{n+1}||_\infty = \frac{2}{n(n+2)}$ for all n, so we have $\sum_{n=1}^{\infty} ||F_n - F_{n+1}||_\infty = \sum_{n=1}^{\infty} \frac{2}{n(n+2)} < \infty$.
In addition, $F_n(x_*^{(n)}) = \frac{1}{n+1} \to 0$, which is the global minimum value of F, but $\{x_*^{(n)}\} = \{1, 0, 1, 0, 1, 0, \ldots\}$ is not a convergent sequence.

But some general conditions may be given to verify the convergence $x_*^{(n)} \to x_*$. Let Ω_F be the set of local minima of F.

Proposition 2. *If every local minimum of F is strict, then $x_*^{(n)}$ is convergent, and its limit verify all previous results.*

Proof. For each $x \in V$, a neighborhood $\mathcal{N}_x$ of x is defined such that, if looking for the minimum starting from point x, every $\hat{x} \in \mathcal{N}_x$ must be checked, and the next point will be the one with the lowest value of the objective function.
Suppose $\Omega_F = \{\xi_1, \ldots, \xi_l\}$. If there only exists one j such that $F(\xi_j) = F(x_*)$, the result is proved, since it will be $x_*^{(n)} \to x_* = \xi_j$.
Suppose that there exist $\{\xi_{j_1}, \ldots, \xi_{j_k}\} \subset \Omega_F$ such that $F(\xi_{j_i}) = F(x_*)$ for all i. Since ξ_j is a strict local minimum, $F(\xi_j) < F(x)$ is obtained for all $x \in \mathcal{N}_{\xi_j}$, and for all j. By Corollary 2, there exists n_0 such that, if $n \geq n_0$, $F_n(\xi_j) < F_n(x)$ for all $x \in \mathcal{N}_{\xi_j}$, and for all j.
Now, for some $n_1 \geq n_0$, it must be $x_*^{(n_1)} = \xi_{j_i}$ for some i, since otherwise it could not be true $F_n(x_*^{(n)}) \to F(x_*)$ and $F(x_*)$ local minimum of F.
But, as stated above, in the $(n_1 + 1)$-th iteration, the next inequality will be verified: $F_{n_1+1}(x_1^{(n_1+1)}) = F_{n_1+1}(x_*^{(n_1)}) = F_{n_1+1}(\xi_{j_i}) < F_{n_1+1}(x)$ for all $x \in \mathcal{N}_{\xi_{j_i}}$, and so $x_*^{(n_1+1)} = \xi_{j_i}$. It can be successively proved that $x_*^{(n)} = \xi_{j_i}$ for all $n \geq n_1$. So $x_*^{(n)} \to x_* = \xi_{j_i}$.

In Example 1, a condition fails: local minima are not strict.
So, in practice, it is guaranteed that x_* is a local minimum of F and, since the sequence $F(x_*^{(n)})$ is not necessarily decreasing, it is expected that this technique escape from certain local minima, improving so the efficiency of the algorithm.

4.2 Application of 'Functional Annealing' to Neural Networks

Analogously to what has been explained in the previous section, since the function to be minimized in case of recurrent neural networks is the energy function E, it can be identified to the objective function F. So, the idea is to look for a sequence E_n (analogue to F_n), verifying the same restrictions as E, defined by Eq. (1), but easier to minimize than E. In fact, every E_{n+1} should be a little more difficult to minimize than the previous E_n.
Although this technique can be easily applied to Hopfield discrete network, it is more useful to get based on the model MREM previously described, since it is a more general model than Hopfield's one, and solutions to many problems, including TSP, are better represented with this multivalued model than with Hopfield's.
The sequence of approximated energy functions we will consider is given by the next expression:

$$E_n(\boldsymbol{S}) = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j}^{(n)} f^{(n)}(s_i, s_j) \tag{2}$$

where $W^{(n)} = (w_{i,j}^{(n)})$ is a sequence of matrices verifying $W^{(n)} \to W$, and $f^{(n)} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is a sequence of functions such that $f^{(n)}(x, y) \to f(x, y)$ for all $x, y \in \mathcal{M}$. Obviously, $E(\boldsymbol{S}) = \lim_{n \to \infty} E_n(\boldsymbol{S})$ for every state vector $\boldsymbol{S} \in \mathcal{M}^N$.
The dynamics used to minimize E_n is exactly the proposed for the MREM model above.
So, for every n, given $\boldsymbol{S}_1^{(n)}$, a sequence $\{\boldsymbol{S}_i^{(n)}\}_{i \geq 1}$, convergent to a vector $\boldsymbol{S}_*^{(n)}$, is obtained.

Theorem 2. *Let $\boldsymbol{S}_1^{(n+1)} = \boldsymbol{S}_*^{(n)}$, $\boldsymbol{S}_1^{(1)}$ randomly chosen, and suppose that these three conditions are satisfied:*

1. $\sum_{n=1}^{\infty} ||W^{(n)} - W^{(n+1)}||_\infty < \infty$
2. $\sum_{n=1}^{\infty} ||f^{(n)} - f^{(n+1)}||_\infty < \infty$
3. $||f^{(n)}||_\infty < K$ *for all* n *(as* $\mathcal{M}^N$ *is discrete, this is always true)*

then the sequences $\{E_n(S_*^{(n)})\}_{n\geq 1}$ *and* $\{E(S_*^{(n)})\}_{n\geq 1}$ *converge to* $E(\boldsymbol{S}_*)$ *for some state vector* $\boldsymbol{S}_* = \lim_{n\to\infty} \boldsymbol{S}_*^{(n)}$.

Proof. Let us verify that Theorem 1 can be applied. To this end, the only thing to be verified is $\sum_{n=1}^{\infty} ||E_{n+1} - E_n||_\infty < \infty$.
Given $\boldsymbol{S} \in \mathcal{M}^N$, it can be proved that (denoting $f^{(n)}(s_i, s_j) = f_{i,j}^{(n)}$)

$$|E_{n+1}(\boldsymbol{S}) - E_n(\boldsymbol{S})| \leq \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} |w_{i,j}^{(n+1)} f_{i,j}^{(n+1)} - w_{i,j}^{(n)} f_{i,j}^{(n)}| \leq$$

$$\leq \frac{1}{2} \sum_{i,j=1}^{N} \left[|w_{i,j}^{(n+1)}||f_{i,j}^{(n+1)} - f_{i,j}^{(n)}| + |w_{i,j}^{(n+1)} - w_{i,j}^{(n)}||f_{i,j}^{(n)}| \right] \leq$$

$$\leq \frac{N}{2} \left[||W^{(n+1)}||_\infty ||f^{(n+1)} - f^{(n)}||_\infty + ||W^{(n+1)} - W^{(n)}||_\infty ||f^{(n)}||_\infty \right]$$

So, taking maximum $||E_{n+1} - E_n||_\infty \leq$

$$\leq \frac{N}{2} \left[||W^{(n+1)}||_\infty ||f^{(n+1)} - f^{(n)}||_\infty + ||W^{(n+1)} - W^{(n)}||_\infty ||f^{(n)}||_\infty \right]$$

But, since $W^{(n)} \to W$, it is obtained that $||W^{(n)}||_\infty \to ||W||_\infty$ and so there exists M such that $||W^{(n)}||_\infty < M$ for all n. And, by condition 3., we have $||f^{(n)}||_\infty < K$ for all n. Then

$$||E_{n+1} - E_n||_\infty \leq \frac{N}{2} \left[M||f^{(n+1)} - f^{(n)}||_\infty + K||W^{(n+1)} - W^{(n)}||_\infty \right]$$

By adding for $n \geq 1$, and applying conditions 1. and 2., we reach at

$$\sum_{n=1}^{\infty} ||E_{n+1} - E_n||_\infty \leq$$

$$\leq \frac{N}{2} \left[M \sum_{n=1}^{\infty} ||f^{(n+1)} - f^{(n)}||_\infty + K \sum_{n=1}^{\infty} ||W^{(n+1)} - W^{(n)}||_\infty \right] < \infty$$

So Theorem 1 and its Corollaries can be applied, and the proof is done.

Corollary 4. *The value $E(\boldsymbol{S}_*)$ is a local minimum of the energy function E.*

It must be noted that when a finite number of functions $\{E_1, \ldots, E_n\}$ is used, the conditions assumed by the previous result are verified, which happens in most applications.

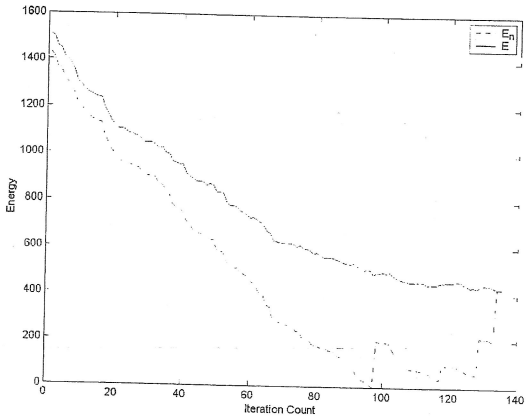

Fig. 1. Energy fluctuations between iterations, comparing the value of E_n (lower graph) and E (upper graph).

5 Experimental Results

For the TSP to be solved only a finite number of modified energy functions is considered: $\{E_1, E_2, \ldots, E_{n_{\text{app}}+1} = E\}$, where all weight matrices are equal to W ($W^{(n)} = W$) and similarity functions are defined by the following expression:

$$f^{(n)}(s_i, s_j) = \begin{cases} 0 & X_{s_i} \in \mathcal{V}_{n_{\text{app}}-n+1}(X_{s_j}) \vee \\ & \vee X_{s_j} \in \mathcal{V}_{n_{\text{app}}-n+1}(X_{s_i}) \\ f(s_i, s_j) = -2d_{s_i,s_j} & \text{otherwise} \end{cases}$$

where $\mathcal{V}_m(X_k)$ is the set formed by X_k and the m cities which are closest to X_k, such that $\mathcal{V}_0(X_k) = \{X_k\}$.

This means that, in the first iterations, the model is trying to build a global order, not focusing in sorting cities close each other, but making a rough order that can be refined in the following iterations, taking in account the distances between neighboring cities.

In fact, this technique is able to avoid certain local minima of the energy function E, as the objective function to be minimized is E_n, allowing the net to increase the value of the original energy function several times, as can be observed in Fig. 1. The border points indicating the step from E_n to E_{n+1} are located in abscissas 97, 117, 128, 134 and 136.

Results of applying this new technique to the classical problems from the TSPLIB library, available on internet and created by Reinelt [16], and directly applying *3-opt* dynamics to minimize E (that is, by using directly MREM to minimize E), are shown in Table 1, where a value of $n_{\text{app}} = 5$ has been considered, over an aggregate of 50 runs for each of the methods for every instance of the problem. No strong dependence on the choice of n_{app} has been observed in the performed simulations.

It can be observed that the technique herein proposed is able to considerably improve the results given by the original MREM model, and those of KNIES, which were improved by MREM, such that the average result in these simulations

Table 1. Comparative results (error percentage over the optimum) of models KNIES, MREM and FA after 50 runs. Note that KNIES always obtains the same results for given parameters, so it needs to fine-tune the parameters to achieve a good solution.

		KNIES	MREM		FA ($n_{\text{app}} = 5$)	
Problem	Opt	Min	Min	Av	Min	Av
eil51	426	2.86	0.23	2.43	0.00	1.42
st70	675	1.51	0.00	1.89	0.00	1.09
eil76	538	4.98	1.30	3.43	0.19	2.74
rd100	7910	2.09	0.00	3.02	0.00	2.41
eil101	629	4.66	1.43	3.51	0.16	2.47
lin105	14379	1.29	0.00	1.71	0.00	0.77
pr107	44303	0.42	0.15	0.82	0.00	0.50
pr124	59030	0.08	0.00	1.23	0.07	0.86
bier127	118282	2.76	0.42	2.06	0.04	1.45

given by this new method is better than that of MREM, showing its ability to escape from local minima. This fact can also be verified when the optimal results of these three methods are compared, since FA outperforms MREM and KNIES in this case too.

It must be noted that the computational time of MREM with FA, with respect to the original MREM model, is not multiplied by $n_{\text{app}} + 1$, since in the last iterations, where n is close to n_{app}, the change in the approximated energy function, when changing from E_n to E_{n+1}, is very little (since the difference from $f^{(n)}$ to $f^{(n+1)}$ is), the network state is very close to a local minimum, so the net hardly iterates when trying to minimize $E_{n_{\text{app}}}$, as shown in Fig. 1. In fact, for $n_{\text{app}} = 5$, FA only last an average of 4 times the time used by MREM, instead of 6 times.

6 Conclusions

In this work we have presented a new general optimization technique that allows to avoid certain local minima, improving considerably the quality of the solutions obtained by several algorithms.

Its theoretical foundations has been proposed, based on very applicable and general enough results, which could be the basis for a more general optimization theory. In addition, its neural application has been presented, showing its possible use in this case.

To test this new technique, the well-known Traveling Salesman Problem has been chosen, since it is the main benchmark for combinatorial optimization algorithms, presenting many methods for its approximated resolution, some of them being very powerful, as MREM or KNIES.

By making some simulations, the improvement supposed by 'Functional Annealing' has been shown, since it is able to avoid many local minima, improving so the quality of the average solution, and reaching, in many cases, solutions very close to the problem optimum.

References

1. Aarts, E.H.L., Korst, J.H.M. and Laarhoven, P.J.M., *A Quantitative Analysis of the Simulated Annealing Algorithm: A Case Study for the Traveling Salesman Problem*, J. Stats. Phys. **50**, 189-206, 1988.
2. Aras, N., Oomen, B.J., and Altinel, I.K., *The Kohonen Network Incorporating Explicit Statistics and its Application to the Travelling Salesman Problem*, Neural Networks, **12**, 1273-1284, 1999.
3. Bland, R.E., and Shallcross, D. F.,*Large Traveling Salesman Problem Arising from Experiments in X-ray Crystallography: a Preliminary Report on Computation*, Technical Report No. 730, School of OR/IE, Cornell University, Ithaca, Nueva York, 1987.
4. Fiechter, C.N., *A Parallel Tabu Search Algorithm for Large Scale Traveling Salesman Problems*, Working Paper 90/1 Department of Mathematics, Ecole Polytechnique Federale de Lausanne, Suiza, 1990.
5. Hopfield, J.J. and Tank, D.W., *Neural Computation of Decisions in Optimization Problems*, Biological Cybernetics, **52**, 141-152, 1985.
6. Kohonen, T., *Self-organizing Maps*, Springer, 1995.
7. Mérida-Casermeiro, E., Galán-Marín, G. and Muñoz-Pérez, J., *An Efficient Multivalued Hopfield Network for the Travelling Salesman Problem*, Neural Processing Letters, **14**, 203-216, 2001.
8. Mérida-Casermeiro, E., Muñoz-Pérez, J., and Domínguez-Merino, E., *An N-parallel Multivalued Network: Applications to the Travelling Salesman Problem*, Computational Methods in Neural Modelling, Lecture Notes in Computer Science, **2686**, 406-41, 2003.
9. Mérida-Casermeiro, E. and Muñoz-Pérez, J., *MREM: An Associative Autonomous Recurrent Network*, Journal of Intelligent and Fuzzy Systems, **12** (3-4), 163-173, 2002.
10. Mérida-Casermeiro, E., Muñoz-Pérez, J. and Benítez-Rochel, R., *Neural Implementation of Dijkstra's Algorithm*, Lecture Notes in Computer Science **2686**, 342-349, 2003.
11. Mérida-Casermeiro, E. and López-Rodríguez, D., *Multivalued Neural Network for Graph MaxCut Problem*, in Proceedings of ICCMSE 2004, **1**, 375-378, 2004.
12. Mérida-Casermeiro, E. and López-Rodríguez, D., *Graph Partitioning via Recurrent Multivalued Neural Networks*, IWANN 2005, Lecture Notes in Computer Science **3512**, 1149-1156.
13. Potvin, J.V., *Genetic Algorithms for the Traveling Salesman Problem*, Annals of Operations Research **63**, 339-370, 1996.
14. Potvin, J.V., *The Traveling Salesman Problem: A Neural Network Perspective*, INFORMS Journal on Computing, **5**, 328-348, 1993.
15. Reinelt, G., *The Travelling Salesman. Computational Solutions for TSP Applications*, Springer, 1994.
16. Reinelt, G., *TSPLIB - A Travelling Salesman Problem Library*, ORSA Journal of Computing, **3**, 376-384, 1991.
17. Wilson, V. and Pawley, G.S., *On the Stability of the TSP Problem Algorithm of Hopfield and Tank*, Biological Cybernetics, **58**, 63-70, 1988.