Stochastic Multivalued Network for Optimization. Application to the Graph MaxCut Problem

Domingo López-Rodríguez[1], Enrique Mérida-Casermeiro[1], J. M. Ortiz-de-Lazcano-Lobato[2]

[1]Department of Applied Mathematics
University of Málaga
Málaga
SPAIN

[2]Department of Computer Science and Artificial Intelligence
University of Málaga
Málaga
SPAIN

Abstract: - The aim of this paper is to present the stochastic version of the multivalued neural model MREM, which has achieved very good results in many applications, as an optimization technique. The purpose of this stochastic version is to avoid certain local minima of the objective function minimized by the network, that is, the energy function. To this end, the description of the theoretical bases of this model, guaranteeing the convergence to minima, is carried out rigorously. In order to show the efficiency of this new model, the model, in its two versions, deterministic and stochastic, has been applied to the resolution of the well-known problem of graph partition, MaxCut. Computational experiments show that in most cases the stochastic model achieves better results than the deterministic one.

Key-Words: Neural Networks, Stochastic Dynamics, Optimization Problems, Graph Problems.

1 Introduction

In classical literature, the MaxCut problem is defined as follows: Given an undirected weighted graph $G = (V, E)$, where $V = \{v_i\}$ is the set of N vertices and E is the set of n_e edges, and edge weights are given by matrix $C = (c_{i,j})_{i,j=1,...,N}$ (meaning that the weight or cost of the edge joining nodes i and j is $c_{i,j} \geq 0$), find a *maximum cut* of G, i.e., a partition of V into two sets that maximizes the total cost of the edges with endpoints in different sets.

This problem arises in the resolution of many practical or theoretical situations. Some examples include: pattern recognition, clustering, statistical physics and the design of communication networks, VLSI circuits and circuit layout [1].

So, this problem is well-known in literature. Due to its wide applicability, many variants of it have been formulated, placing restrictions on the original formulation.

The original problem, with all the variants, is known to be NP-complete [2], making their resolution computationally intractable, but in the case of planar graphs they belong to P, that is, there exists a solution in polynomial time. So, many algorithms have appeared to tackle MaxCut in the general case.

In 1997, Alberti et al. presented a Hopfield-like neural model for MaxCut [3], but its performance is worse than the presented by Bertoni et al [4]. Takefuyi and his colleagues [5] developed a powerful neural model named 'maximum' and it proved to perform better than the rest of algorithms in solving a wide range of combinatorial optimization problems.

In the last few years, Galán-Marín et al. [6] proposed a new neural model named OCHOM which obtains much more efficient solutions than 'maximum'. Moreover, it can be used for many problems and it also has the advantage of fast convergence to a valid

solution without tuning any parameter. In order to make OCHOM escape from local minima, Wang et al. [7] have recently proposed a stochastic dynamics for OCHOM, permitting temporary decreases of the objective function.

Note that there exists very few bibliographic references for K-partitioning (most of the references is focused in bipartition).

Recently, Mérida et al. [8] presented a neural model called MREM that has the ability of K-partitioning the graph, since it is a multivalued model. This model had been very successful in other combinatorial optimization problems, see for example [9, 10, 11, 12].

The aim of this work is to present the stochastic version of this model MREM that helps to escape from certain local minima, improving so its efficiency when dealing with problems presenting difficulties, like the one studied in this paper.

2 Formal Description of the Problem

Let $G = (V, E)$ be an undirected graph without self-connections. $V = \{v_i\}$ is the set of vertices and E is the set of n_e arcs. For each edge in E there is a weight $c_{i,j} \in \mathbb{R}^+$. All weights can be expressed by a symmetric real matrix C, with $c_{i,j} = 0$ when it does not exist an arc with endpoints v_i and v_j.

The Maximum Cut Problem (MaxCut): consists in finding a partition of V into two subsets A_1 and A_2, such that $\sum_{v_i \in A_1, v_j \in A_2, i>j, m \neq n} c_{i,j}$ is maximum.

Generalization of the MaxCut Problem (K-MaxCut): It looks for a partition of V into K disjoint sets A_i such that the sum of the weights of the edges from E that have their endpoints in different elements of the partition is maximum. So, the function to be maximized is

$$\sum_{v_i \in A_m, v_j \in A_n, i>j} c_{i,j} \tag{1}$$

3 The Stochastic MREM model

Let us remember that the deterministic 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 $\mathcal{M}$.

Network state is completely determined by a state vector $\vec{S} = (s_1, s_2, \ldots, s_N) \in \mathcal{M}^N$, 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:

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

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.

The purpose of the deterministic net is to minimize the energy function described before. To this end, a random initial state $\vec{S}_0$ is introduced into the net and at time t, state vector $\vec{S}(t)$ will be changed for another state vector $\vec{S}(t+1)$ (defined by the computational dynamics) if $E(\vec{S}(t+1)) < E(\vec{S}(t))$. If there is not any $\vec{S}'$ in the neighbourhood of $\vec{S}(t)$ such that $E(\vec{S}') < E(\vec{S}(t))$, then $\vec{S}(t)$ is a local minimum (for the considered dynamics) and the net stops iterating.

The stochastic model sMREM has the same architecture as the deterministic one. It is based in the same energy function, given by Eq. (2), but the decreasing of energy is not guaranteed. In fact, it depends on a sequence $\{T_n\}$, analogous to the temperature sequence in Simulated Annealing [13].

This new model builds a sequence of state vectors $\{S_*^{(n)}\}$, where the super-index (n) shows that in this state the 'temperature' of the net was T_n.

The dynamics consists in:

- $S_1^{(1)}$ is randomly generated.
- Given a state of the net, $\vec{S} = \vec{S}_m^{(n)}$, another state vector $\vec{S}'$ is randomly sampled from a neighbourhood $\mathcal{N}_{\vec{S}}$ of $\vec{S}$.
- The increment of energy corresponding to updating the net from $\vec{S}$ to $\vec{S}'$ is computed: $\Delta E = E(\vec{S}') - E(\vec{S})$.
- Then, the net accepts the next state of the net $\vec{S}_{m+1}^{(n)} = \vec{S}'$ with probability $\mathbb{P}(\Delta E)$, depending on the value of T_n.
- If $m + 1 = M$, then define, for simplicity, $\vec{S}_*^{(n)} = \vec{S}_M^{(n)}$, increment the value of n and $m = 1$. Otherwise, increment the value of m.

A pair of conditions must be satisfied in order to guarantee convergence to states of minimal energy:

1. $\lim_{n\to\infty} \mathbb{P}\left(E(\vec{S}_{m+1}^{(n)}) > E(\vec{S}_m^{(n)})\right) = 0$ for all $m \in \{1,\ldots,M-1\}$

2. The acceptation probability of $\vec{S}'$ as $\vec{S}_{m+1}^{(n)}$ is of the form

$$\mathbb{P}(\Delta E) = \begin{cases} 1, & \text{if } \Delta E < 0 \\ g_n(\Delta E) < 1, & \text{if } \Delta E \geq 0 \end{cases}$$

where $g_n\colon \mathbb{R}^+ \to [0,1)$ and $\Delta E = E(\vec{S}') - E(\vec{S}_m^{(n)})$. In addition, it must be $g_n(\Delta E) > 0$ for all ΔE. Note that the definition of g_n depends on the temperature T_n.

It can be proved that if $\{g_n\}$ converges uniformly to 0, then the second condition implies the first one.

Next we describe the theoretical basis of this model that guarantees the convergence to states of minimal energy:

Theorem 1 *With probability 1, there exists $L \in \mathbb{R}$ such that $L = \lim_{n\to\infty} E(\vec{S}_*^{(n)})$.*

We omit the proof of this theorem because of the limitation on the length of the present paper.

Corollary 2 *If $\hat{\vec{S}}$ is an accumulation point of the sequence $\{\vec{S}_*^{(n)}\}$, then, with probability 1, we have $E(\hat{\vec{S}}) = \lim_{n\to\infty} E(\vec{S}_*^{(n)})$.*

We will accept without demonstration, due to the limitation on the length of this paper, the following lemma:

Lemma 3 *The next equality holds $\lim_{n\to\infty} \mathbb{P}\left(E(\vec{S}_m^{(n)}) \geq E(\vec{S}_*^{(n)})\right) = 1$ for all $m \in \{1,\ldots,M-1\}$.*

By making use of this lemma, we can prove the following theorem:

Theorem 4 *Let $\vec{S}$ be a state vector with $E(\vec{S}) < L$, where*

$$L = \lim_{n\to\infty} E(\vec{S}_*^{(n)})$$

Then, the probability of sampling $\vec{S}$ is 0 for all $n \geq N$.

Proof.
We know that $L = \lim_{n\to\infty} E(\vec{S}_*^{(n)})$ with probability 1 if, and only if, for all $\varepsilon > 0$ there exists $n_0 \in \mathbb{N}$ such that if $n \geq n_0$ then $|E(\vec{S}_*^{(n)}) - L| < \varepsilon$ with probability 1, that is, $E(\vec{S}_*^{(n)}) \in (L-\varepsilon, L+\varepsilon)$ with probability 1.

Let us consider $\varepsilon = \frac{L-E(\vec{S})}{2}$. Then, $E(\vec{S}) < L - \varepsilon = E(\vec{S}) + \varepsilon$.

Let $n \geq n_0$.

Let us suppose that the probability of sampling the vector $\vec{S}$ is $\rho > 0$.

We can consider the set $A = \{E(\vec{S}_*^{(n+1)}) \notin (L-\varepsilon, L+\varepsilon)\}$ and compute a bound for its probability:

$$\mathbb{P}(A) \geq \mathbb{P}(\textit{sampling } \vec{S}) \cdot \mathbb{P}(\text{accepting } \vec{S}_m^{(n+1)} = \vec{S}) \cdot$$
$$\cdot \mathbb{P}\left(E(\vec{S}_m^{(n+1)}) \geq E(\vec{S}_*^{(n+1)})\right)$$

The first of these 3 probabilities is equal to ρ. The third one has limit 1 as n tends to ∞ (by the previous lemma), so there exists n_1 such that if $n \geq n_1$ then that probability is greater than $\eta > 0$. The second one is always positive, since we imposed the hypothesis of $g_n(\Delta E) > 0$ for all n. In particular, it is positive for all $n \geq n_1$.

Let us take $N = \max\{n_0, n_1\}$ and $n \geq N$. Then we arrive at $\mathbb{P}(A) \geq \rho\eta\mu > 0$, where $\mu = \mathbb{P}(\text{accepting } \vec{S}_m^{(n+1)} = \vec{S})$. Therefore, we have that the probability $\mathbb{P}\left(E(\vec{S}_*^{(n+1)}) \in (L-\varepsilon, L+\varepsilon)\right) = 1 - \mathbb{P}(A) < 1 - \rho\eta\mu < 1$, what contradicts that $E(\vec{S}_*^{(n)}) \in (L-\varepsilon, L+\varepsilon)$ with probability 1 for all $n \geq n_0$.

As conclusion, the probability of sampling the state vector $\vec{S}$ is 0. □

This result provides two important Corollaries dealing with the optimality of the accumulation points of $\{\vec{S}_*^{(n)}\}$ and the convergence of this sequence.

Proposition 5 *Let $\hat{\vec{S}}$ be an accumulation point of the sequence $\{\vec{S}_*^{(n)}\}$. Then, $E(\hat{\vec{S}}) \leq E(\vec{S})$ for all $\vec{S} \in \mathcal{N}_{\hat{\vec{S}}}$. So, $\hat{\vec{S}}$ is a local minimum of E.*

Proof.
The proof is an immediate consequence of the previous theorem. □

For the next result, we will need that $\{g_n\}$ converges uniformly to 0, that is, we will need that $\lim_{n\to\infty} ||g_n||_\infty = 0$ where $||g_n||_\infty = \sup_{t\in\mathbb{R}^+} |g_n(t)|$.

Proposition 6 *If local minima of E are strict, $\{g_n\}$ converges to 0 uniformly, and*

$\hat{\vec{S}}$ is an accumulation point of $\{\vec{S}_*^{(n)}\}$, then $\lim_{n\to\infty} \mathbb{P}\left(\vec{S}_*^{(n+1)} = \hat{\vec{S}} | \vec{S}_*^{(n)} = \hat{\vec{S}}\right) = 1$.

Proof.

Let us consider n such that $\vec{S}_*^{(n)} = \hat{\vec{S}}$ (there exists an infinite number of n's verifying that condition, since $\hat{\vec{S}}$ is an accumulation point of the sequence $\{\vec{S}_*^{(n)}\}$). Let us compute the probability of $\{\vec{S}_*^{(n+1)} \neq \hat{\vec{S}}\}$:

$$\mathbb{P}(\vec{S}_*^{(n+1)} \neq \hat{\vec{S}}) =$$

$$= \mathbb{P}(\text{sampling } \vec{S} \in \mathcal{N}_{\hat{\vec{S}}}) \cdot \mathbb{P}(\text{accepting } \vec{S} = \vec{S}_2^{(n+1)}) \cdot$$

$$\cdot \mathbb{P}(\text{no return to } \hat{\vec{S}}) \leq \mathbb{P}(\text{accepting } \vec{S} = \vec{S}_2^{(n+1)})$$

The probability of accepting $\vec{S} = \vec{S}_2^{(n+1)}$ starting from $\hat{\vec{S}} = \vec{S}_1^{(n+1)}$ is

$$\mathbb{P}(\Delta E) = g_n(\Delta E) \leq ||g_n||_\infty$$

whose limit is 0. Therefore, $\mathbb{P}(\vec{S}_*^{(n+1)} \neq \hat{\vec{S}} | \vec{S}_*^{(n)} = \hat{\vec{S}})$ tends to 0.

From this fact we can conclude that

$$\lim_{n\to\infty} \mathbb{P}\left(\vec{S}_*^{(n+1)} = \hat{\vec{S}} | \vec{S}_*^{(n)} = \hat{\vec{S}}\right) =$$

$$= 1 - \lim_{n\to\infty} \mathbb{P}\left(\vec{S}_*^{(n+1)} \neq \hat{\vec{S}} | \vec{S}_*^{(n)} = \hat{\vec{S}}\right) = 1$$

and the proof is complete. □

So we have proved that the stochastic MREM model, with the sampling-accepting scheme developed in this section, is able to converge to a local minimum of the energy function E.

In addition, we have proved that this convergence is not dependent on the rate of convergence of the temperature sequence $\{T_n\}$, it only depends on the convergence of $||g_n||_\infty$ to 0.

4 Application of the Model to MaxCut Problem

In order to solve the MaxCut problem with this neural net, we need as many neurons as number of nodes N in the graph. Each neuron taking value $s_i \in \mathcal{M} = \{1, 2, \ldots, K\}$ points to the subset of the partition where the i-th node is assigned to.

The cost function of the K-MaxCut problem, given by Eq. (1), must be identified with the energy function of Eq. (2). So, for MaxCut, it is $w_{i,j} = c_{i,j}$, and $f(x, y) = \delta_{x,y}$ (Krönecker delta function), also valid for K-MaxCut, since it is equivalent to maximize the cost of the edges cut by the partition and to minimize the cost of the edges whose endpoints lie within the same group of the partition.

In this work, a simple dynamics, named best-2, has been firstly implemented.

best-2: It consists in getting the greatest decrease of the energy function just by changing the state of only two neurons at each time. So, a set of neighboring states must be defined. If neurons to be changed are p and q, this set will be named $\mathcal{N}_{p,q}$. Then, if $\vec{S}(t)$ is the state of the net at time t, $\vec{S}(t+1)$ will be the vector from a $\mathcal{N}_{p,q}$ that maximizes the decrease of energy, $-\Delta E$. In the case of this problem, the neighbourhood $\mathcal{N}_{p,q}$ of $\vec{S}$ include all possible states from $\mathcal{M}^N$ that differ from $\vec{S}$ only in the outputs of neurons p and q (or both). So, there will be K^2 vectors in $\mathcal{N}_{p,q}$.

An expression for the decrease of energy is here given in order to reduce the computational cost of the model. Suppose that neurons p and q are going to be changed, and that we denote $s_i(t) = s_i$ and $s_i(t+1) = s_i'$ for all i. Then, the decrease of energy is given by $U_{p,q} = -\Delta E =$

$$= \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{i,j} \left(f(s_i, s_j) - f(s_i', s_j') \right)$$

$$= \sum_{i=1}^{N} (\Delta_{i,p} + \Delta_{i,q}) - \Delta_{p,q} \qquad (3)$$

(provided the symmetry of function f), where $\Delta_{i,j} = w_{i,j} \left(f(s_i, s_j) - f(s_i', s_j') \right)$.

So, the dynamics best-2 can be summarized as follows:

1. A state for the net is initially randomly assigned.
2. Repeat until no change in state vector:
   (a) The scheduling selects a value $d \in \{1, \ldots, \lfloor \frac{N}{2} \rfloor\}$. For $d > \lfloor \frac{N}{2} \rfloor$, all of the following computations are made twice, and this way we can save some computational effort.
   (b) The following can be made parallel: every neuron p studies all possibilities of changing neurons p and $q = (p + d) \mod (N)$, with $0 < q \leq N$, i.e., p computes the potential associated to the possible changes, it is stored

as a vector $\vec{u_p}$ whose components are the decrease of energy associated to any vector in $\mathcal{N}_{p,q}$, by applying (3).

(c) Neuron p computes $\vec{\alpha}(p) = \max \vec{u_p}$, associated to a state $\vec{\vec{S}}_{p,q} \in \mathcal{N}_{p,q}$.

(d) The scheduling selects the next state of the net, $\vec{S}(t+1) = \vec{\vec{S}}_{p,q}$ for which $p = \arg\max \vec{\alpha}$.

Some experimental results for the dynamics herein proposed are shown in the next section.

5 Experimental Results

In this section we will show the experimental results of comparing both deterministic and stochastic versions of the mutivalued model MREM.

A test set was formed by 280 random graphs depending on two parameters, $N \in \{50, 100, 150, 200\}$ (the cardinality of the set of vertices), and the second being $\rho \in \{0.05, 0.15, 0.25, 0.5, 0.75, 0.9\}$ (the density of edges in the graph, meaning that $n_e \approx \rho\frac{N(N-1)}{2}$). Weights for edges were integers randomly chosen in $[0, 10]$. For this set to be complete, the values for the parameters were chosen to cover a wide range of graphs.

For each graph, 10 independent executions were performed. MREM was used with the dynamics named best-2, and sMREM used the stochastic version of best-2, with

$$g_n(\Delta E) = \exp\left(\frac{-|\Delta E|}{T_n}\right)$$

(which converges uniformly to 0), the function of acceptation defined in a previous section. In this case, we have only considered a finite number of temperatures $T_1, \ldots, T_{n_a}$, decreasing lineally from $T_1 = 1$ to $T_{n_a} = 0$.

Table 1 presents the results of these experiments. Showing that in most cases (above 70%) the average result provided by sMREM is higher than that of MREM. Its only drawback is the high consumption of time.

6 Conclusions

In this work we have presented the stochastic version of a multivalued neural model called MREM, very useful in many combinatorial optimization problems, with the aim of helping MREM to avoid certain local minima of the energy function.

We have proposed the theoretical bases of this new model, based on results proving its convergence to minima of the objective function.

In order to show the effectiveness of this stochastic version, we have applied these two models to the Max-Cut problem, very well known from the specialized literature and because of its applications, since MREM has recently achieved the best results ever obtained by a neural model. These experiments have shown that in most cases sMREM outperforms MREM, but by using more computational time. The reduction in the computational time used by sMREM is an issue to be studied as a future research line.

References:

[1] Barahona, F., Grotschel, M., Junger, M., Reinelt, G.: An application of combinatorial optimization to statistical physics and circuit layout design. Operat. Research **36** (1988) 493 – 513

[2] Garey, M., Johnson, D.: Computers and Intractability. A guide to the theory of NP-Completeness. W. H. Freeman and Company (1979)

[3] Alberti, A., Bertoni, A., Campadelli, P., Grossi, G., Posenato, R.: A neural algorithm for max-2sat: performance analysis and circuit implementation. Neural Networks **10** (1997) 555–560

[4] Bertoni, A., Campadelli, P., Grossi, G.: An approximation algorithm for the maximum cut problem and its experimental analysis. In: Algorithms and Experiments. Volume 9. (1998) 137 – 143

[5] Takefuyi, Y., Wang, J.: Neural computing for optimization and combinatorics. Volume 3. World Scientific (1996)

[6] Galán-Marín, G., Muñoz-Pérez, J.: Design and analysis of maximum hopfield networks. IEEE Trans. on Neural Networks **12** (2001) 329 – 339

[7] Wang, J., Tang, Z.: An improved optimal competitive hopfield network for bipartite subgraph problems. Neurocomputing **61** (2004) 413 – 419

[8] Mérida-Casermeiro, E., López-Rodríguez, D.: Graph partitioning via recurrent multivalued neural networks. Lecture Notes in Computer Science **3512** (2005) 1149 – 1156

[9] Mérida-Casermeiro, E., Galán-Marín, G., Muñoz Pérez, J.: An efficient multivalued hopfield network for the travelling salesman problem. Neural Processing Letters **14** (2001) 203 – 216

		MREM			sMREM		
N	ρ	Best	Av.	t	Best	Av.	t
50	0.05	276.8	256.28	0.05	276.8	256.8	1.31
50	0.15	672.8	631.56	0.06	687.2	637.2	1.37
50	0.25	1013.2	970.84	0.06	1020.4	971.24	1.32
50	0.5	1778.8	1724.08	0.06	1774.4	1729.26	1.29
50	0.75	2663.6	2475.48	0.05	2646.4	2476.92	1.34
50	0.9	2941.8	2876.18	0.06	2949.2	2883.94	1.33
100	0.05	990.2	917.72	0.15	971.2	920.68	5.58
100	0.15	2384.4	2323.9	0.16	2408.6	2340.46	5.54
100	0.25	3719.2	3620.9	0.14	3720.8	3646.9	5.28
100	0.5	6711.6	6637.08	0.13	6745.6	6683.9	5.41
100	0.75	9816.2	9524.1	0.14	9813.4	9581.22	5.21
100	0.9	11348.8	11215.06	0.14	11434.4	11254.66	5.29
150	0.05	2009.8	1933.6	0.26	2020.2	1939.86	14.48
150	0.15	5136	5014.62	0.26	5120.4	5066.62	13.85
150	0.25	7990	7807.16	0.26	8068.2	7868.72	12.84
150	0.5	14701.4	14531.06	0.24	14818	14661.4	12.61
150	0.75	21126.2	20899.94	0.22	21308.6	21018.06	12.58
150	0.9	24926	24589.62	0.22	25056	24729.68	12.05
200	0.05	3411.4	3321.84	0.38	3443.8	3340.26	27.58
200	0.15	8765.4	8653.52	0.42	8873.6	8726.9	25.51
200	0.25	13741	13533.9	0.35	13959.4	13707.9	24.24
200	0.5	25750.8	25500.18	0.34	25967.2	25740.64	22.41
200	0.75	37038.6	36789.2	0.32	37217.8	37063.42	22.04
200	0.9	43584.8	43296.26	0.33	43854.8	43527.38	23.17

Table 1: Comparison results between MREM and sMREM for the 2-MaxCut problem.

[10] Mérida-Casermeiro, E., Muñoz Pérez, J., Benítez-Rochel, R.: Neural implementation of dijkstra's algorithm. Lecture Notes in Computer Science **2686** (2003) 342 – 349

[11] Mérida-Casermeiro, E., López-Rodríguez, D.: Multivalued neural network for graph maxcut problem. In: Lecture Series on Computer and Computational Sciences 2004. Volume 1. (2004) 375 – 378

[12] López-Rodríguez, D., Mérida-Casermeiro, E.: Matrix bandwidth minimization: A neural approach. In: Lecture Series on Computer and Computational Sciences 2004. (2004) 324 – 327

[13] Kirkpatrick, S., Gellat, C.D., Vecchi, M.: Optimization by simulated annealing. Science **220** (1983) 671 – 680