The Continuous Hopfield Networks (CHN) for the Placement of the Electronic Circuits Problem

M. ETTAOUIL K. ELMOUTAOUAKIL Y. GHANOU

UFR: Scientific calculation and Computing, Engineering sciences
Department of Mathematics and Computer science, Faculty of Science and Technology
University Sidi Mohammed ben Abdellah
Box 2202 Fez MOROCCO

mohamedettaouil@yahoo.fr karimmoutaouakil@yahoo.fr yghanou2000@yahoo.fr

Abstract: The Placement of the Electronic Circuits Problem (PECP) is considered as one of the most difficult optimization problems. The PECP has been expressed as a Quadratic Knapsack Problem (QKP) with linear constraints. The goals of this work are to solve the Placement of the Electronic Circuits Problem (PECP) using the Continuous Hopfield Networks (CHN) and to illustrate, from a computational point of view, the advantages of CHN by its implement in the PECP. The resolution of the QKP via the CHN is based on some energy or Lyapunov function, which diminishes as the system develops until a local minimum value is obtained. The Decomposition approach was used to solve the PECP. This method suffers from problems of feasibility of solutions and long training time. Unlike the decomposition approach, the CHN is much faster and all the solutions are feasible. Finally, some computational experiments solving the PECP are included.

Key-Words: Placement of the Electronic Circuits Problem (PECP), Continuous Hopfield Networks (CHN), Quadratic Knapsack Problem (QKP), combinatorial problems, satisfaction of the PECP constraints.

1 Introduction

The aims of this work are to solve the Placement of the Electronic Circuits Problem (PECP) using the Continuous Hopfield Networks (CHN) and to illustrate, from a computational point of view, the advantages of CHN by its implement in the PECP. The PECP is introduced as a QKP. In the last four decades, many researchers working on QKP have proposed methods for linearizing the quadratic term in the objective function by introducing additional variables. The work of Lawler [26] is a fundamental linearization, deriving the well-known *Gilmore-Lawler-Bound* (GLB) and an entire family of correlated linearizations. The research of Kaufman and Broeckx [23], Frieze and Yadegar [12] and more recently Adams and Jonson [19] is extremely important on this matter. The linearization of Adams and Johnson dominates all the others [19]. Closely related to some linearizations are the polyhedral studies performed by Barvinok [11], Jünger and Kaibel [4,2], designed to derive the QKP *polytope* for use with *branch-and-cut* methods. But with these linearizations, the number of the variable and the number of constraints grow dramatically. On the other hand, many authors have chosen to work with QKP in its original form. For example, Finke et al. [4] used the trace formulation to introduce the eigenvalue bounds, a stronger class of lower bounds when compared to GLB bounds. The eigenvalue lower levels of the *Branch-and-bound* tree are searched. Since QKP in NP-Hard, good lower bounds are of eminent importance for solving these problems by implicit enumeration procedures like *Branch-and-bound.*

The PECP had already been treated using the decomposition approach which decomposes an instance of the problem recursively into several small instances. In addition, this mentioned approach keeps a global sight of the entire problem instance at all stages; see Jünger and all [20]. However, this method suffers from problems of feasibility of solutions and long training time.

The CHN was proposed by Hopfield and Tank [16] to solve any combinatorial problem; since it proved its efficiency, many researches have also treated the QKP through this neuronal approach [3],[14],[29]. Within these papers, the feasibility of the equilibrium points of the CHN is not assured for the

general case, and the solutions obtained are often not good enough. To avoid these problems, we choose an appropriate energy function that penalizes the objective function and the constraints of the PECP. The penalty parameters of this function should ensure an appropriate balance between minimization of the cost function and simultaneous satisfaction of the PECP constraints. In addition, these parameters must avoid some bad local minima. In order to realize these goals, we decompose the set of non feasible solutions into appropriate subsets, and we use the partial derivatives of the energy function to select the parameters of the function.

In this work, we formulate the PECP in a manner that can be solved by the CHN; that is, we cast this problem in terms of an appropriate energy function that can be minimized by the CHN.

The paper consists of seven sections. The second introduces the CHN and its dynamical equations. In the third section, the PECP is introduced as a QKP. The use of the decomposition method is presented in the fourth section. In the fifth section, the energy function related to the PECP is determined, where the calculating procedure of the parameter-setting of the energy function is also given. Finally, some computational experiments are included in the sixth section.

2 The Continuous Hopfield neural Networks (CHN)

The CHN consists of n interconnected neurons with a smooth sigmoid activation function (usually a hyperbolic tangent).

The differential equation which governs the dynamics of the (CHN) is:

$$\frac{du}{dt} = -\frac{u}{\tau} + T\,v + i^b \qquad (1)$$

with

$u = (u_1, \ldots, u_n)^t$,

$v = (v_1, \ldots, v_n)^t$,

$v_i = g(u_i), \quad \forall i, j = 1, \ldots, n$,

$g(u_i) = \frac{1}{2}(1 + \tanh(u_i / u_0)), u_0 > 0$,

$T = (T_{ij})_{i,j=1,\ldots,n}$,

$i^b = (i_i^b)_{i=1,\ldots,n}$,

where

u_i : is the current state of the neuron i,

v_i : is the output of the neuron i,

T_{ij} : is the weight of the synaptic connection from neuron j to neuron i,

i_i^b : is the offset bias of the neuron i.

Definition 1

A point u^e is called an equilibrium point of the system (1) if, for an initial input vector u^0, u^e satisfies $u(t) = u^e \quad \forall t \geq t_e$, for some $t_e \geq 0$.

Hopfield has introduced the energy function E on $[0,1]^n$ which is defined by

$$E(v) = -\frac{1}{2} v^t T v - (i^b)^t v + \frac{1}{\tau} \sum_{i=1}^{N} \int_0^{v_i} g^{-1}(x)dx \qquad (2)$$

It should be noted that if the energy function (or Layapunov function) exists, the equilibrium point exists as well. Hopfield proved that the symmetry of matrix T is a sufficient condition for the existence of Lyapunov function; see[17].

The CHN will solve combinatorial problems that can be expressed as the constrained minimization of:

$$E(v) = -\frac{1}{2} v^t T v - (i^b)^t v \qquad (3)$$

The extremes of this function are among the corners of the n-dimensional hypercube $[0,1]^n$; see [15] and [27].

The philosophy of this approach is that the objective function which characterizes the combinatorial problem is associated with the energy function of the network when $\tau \to \infty$. In this way, the output of the CHN can be represented as a solution to combinatorial problem.

Since the differential equation, which characterizes the dynamics of the CHN, is analytically hard to solve, many researchers used to make the use of the famous Euler method. However, this latter proved to be highly sensitive with respect to initial conditions, and it requires a lot of CPU time for medium or greater size CHN instances. Recently, one algorithm, which assures the obtaining of the equilibrium points, has been proposed. Unlike the

former method, the proposed algorithm is robust with respect to the initial conditions; see [28].

Given the combinatorial optimisation problem with n variables and m linear constraints:

$$Min\left\{\frac{1}{2}v^t Pv + q^t v\right\}$$

Subject to

$$Rv = b \qquad (4)$$

$$v_i \in \{0,1\} \quad i = 1,\ldots,n$$

To simplify, the sets are defined as:

● The Hamming hypercube

$$H = [0,1]^n$$

● The Hamming hypercube corners set

$$H_C = \{0,1\}^n$$

● The feasible solutions set

$$H_F = \{v \in H_c \,/\, Rv = b\}$$

The standard form of the energy function is:

$$E(v) = E^c(v) + E^p(v) \quad \forall v \in H$$

where

● $E^c(v)$ is directly proportional to the objective function.

● $E^p(v)$ is a quadratic function that ensures the feasibility of the solution obtained by the CHN, and also penalizes the violated constraints of the problem. This function must give the same value for each element v form H_F, and an adequate selection of this function is necessary for a correct mapping.

3 The Placement of the Electronic Circuits Problem (PECP)

From the computational point of view, the PECP is considered as one of the most difficult optimisation problems; see [20], [24] and [25].

An integrated circuit consists of the central part and the border. The central part is a rectangular array (the master) of base cells and the border consists of two columns of cells; see Fig.1. At the beginning, the circuit integer connections are unknown.

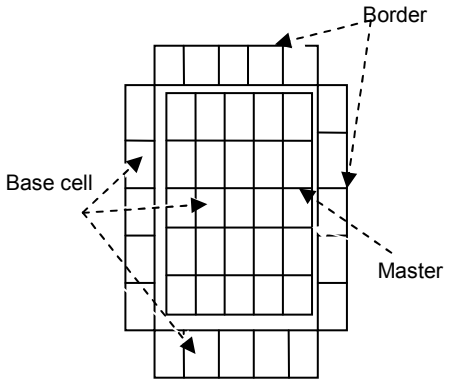

Fig.1: The master and the border consists of base cells.

Let B be a finite set of cells where each cell b has a finite number of rectangular realisations. Each cell is characterized by its widths and heights, the positions of its electrical terminals (pins), and its electrical properties, like switching speed, etc. Moreover, we have a set of nets, and each net $\{b_1,\ldots,b_k\} \subseteq B$ specifies that certain pins of the cells $b_1,\ldots,b_k$ have to be electrically connected.

The PECP looks for a nonoverlapping (Fig.2) assignment of the cells to the plan and realization of the nets with respect to some criteria as: minimal area of the smallest rectangle containing all cells, minimal switching time of the circuit and wire ability with minimum total wire length.

Let m be the number of the base cells of the master and n be the number of the cells.

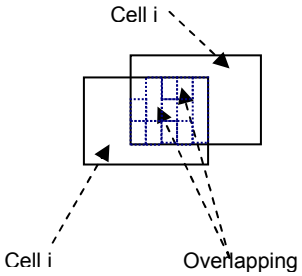

Fig. 2: The common part between the cells i and j represents the overlapping between these two cells.

Definition 2

A cell i is *assigned* to base cell *k* if cell *i* is placed on the master so that its lower left corner coincides with base cell *k*; see Fig.3.

Definition 3

We say that a base cell *k* is feasible for a cell *i* if cell i fits on the master when assigned to base cell *k*.

Definition 4
The Manhattan distance between cells i and j assigned to base cells k and l, respectively, is the sum of the shortest distances in both horizontal and vertical directions between any two points on the boundary of i and j.

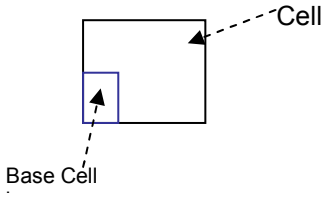

Fig.3: The cell i is assigned to base cell k

Definition 5
The *overlapping number* between two cells is the number of common base cells between these two cells, see Fig.2.

Definition 6
The *density of circuits* is the total area of the cells to be placed, divided by the area of the master.

3.1 The PECP as a 0-1 mathematical programming model

In this part, we introduce an optimization model for the placement of electronic circuit problem based on a quadratic $0-1$ programming problem with special linear constraints.

• Following the notation of M. Jünger and all [20], let v be the $n \times m$ matrix of state variable:

$$\forall (i,k) \in \{1,\ldots,n\} \times \{1,\ldots,m\}$$

$$v_{i,k} = \begin{cases} 1 & if \ \ cell\, i \ is \ assigned \\ & to \ feasible \ \ base \ cell \ k \\ 0 & otherwise \end{cases}$$

This matrix is converted to a $n \times m - vector$:

$$v = \begin{pmatrix} v_{1,1} \\ \vdots \\ v_{1,n} \\ \vdots \\ v_{i,1} \\ \vdots \\ v_{i,n} \\ \vdots \\ v_{n,1} \\ \vdots \\ v_{n,m} \end{pmatrix}$$

• In practice, it is impossible to avoid the overlapping between the cells placed into the "master", and their relative weights have to be controlled by some parameters.
The cost matrix P is defined as the following $\forall i, j = 1,\ldots,n$ and $\forall k, l = 1,\ldots,m$

$$p_{ik,jl} = c_{ij} d_{ik,jl} + \lambda o_{ik,jl} \quad ,$$

where

C_{ij} is the number of nets connecting cells i and j,

$d_{ik,jl}$ is the Manhattan distance between cells i and j,

$O_{ik,jl}$ is the number of overlapping units between cells i and j,

$\lambda \geq 0$ is a penalty parameter.

In general, the matrix d of the Manhattan distance, the matrix O of overlapping numbers and the matrix C of number of nets are symmetric. Therefore, the matrix P is symmetric too.
The objective function is defined by:

$$f(v) = \frac{1}{2} v^t P v + q^t v$$

where

$$q = \underbrace{(0\ldots\ldots0\ldots\ldots0)}_{n \times m}^{t}$$

• All cells must be assigned to exactly one of the m base cells and thus:

$$\sum_{k=1}^{m} v_{i,k} = 1 \qquad \forall\, i \in \{1,\ldots,n\} \qquad (5)$$

In this case, the matrix which defines the constraints is:

$$R=\begin{pmatrix} 1\dots1\dots1 & 0\dots0\dots0 & \dots & 0\dots0\dots0 \\ \vdots \quad \vdots \quad \vdots\vdots & \vdots \quad \vdots & \dots & \vdots \quad \vdots \\ 0\dots0\dots0 & 1\dots1\dots1 & \dots & 0\dots0\dots0 \\ \vdots \quad \vdots \quad \vdots\vdots & \vdots \quad \vdots & \dots & \vdots \quad \vdots \\ 0\dots0\dots0 & 0\dots0\dots0 & \dots & 1\dots1\dots1 \end{pmatrix}$$

and the vector is

$$b=\begin{pmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

Now the Placement of the Electronic Circuits Problem (PECP) is

minimize the cost function:

$$f(v)=\frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{m}\sum_{j=1}^{n}\sum_{l=1}^{m} p_{ij}\, v_{i,k}\, v_{j,l}$$

subject to the blocking constraints

$$\sum_{k=1}^{m} v_{i,k}=1 \qquad \forall i\in\{1,\dots,n\}$$

$$v_{i,k}\in\{1,0\} \qquad \forall i,j=1,\dots,n \quad \forall k,l=1,\dots,m$$

which can be written more compactly in a matrix form

$$Min \quad \left\{\tfrac{1}{2}\, v^{t} P v\right\}$$

Subject to

$$Rv = b$$

$$v\in\{0,1\}^{n\times m}$$

Many researches have solved the QKP using the Lagrangean duality (for example: Lagrangean decompositions methods, Lagrangean substitutions …etc..); see [7]. But these methods are based on gradient descending technique which needs an extremely high number of iterations for training.

4 Decomposition approach for the placement of the electronic circuits problem

In 1991, the decomposition approach was used to solve the PECP; see [15]. This method decomposes the problem recursively into several small instances. In addition, it keeps a global sight of the entire problem instance at all stages. In this section, we will briefly discuss the major headlines of this method.

4.1 The decomposition approach

The decomposition approach consists of the creation of a k-nary tree T in order to decompose a large instance into smaller ones. All cells and the whole area of the circuit construct the root of T. For any node t of the tree, its k-th children $t_1, t_2, \dots, t_k$ corresponding to k rectangular sub-areas $a_{t_1}, a_{t_2}, \dots, a_{t_k}$ of rectangle a_t represented by t, and k subsets $c_{t_1}, c_{t_2}, \dots, c_{t_k}$ of the cell set c_t assigned to node t.

In the following, we will explain how the decomposition scheme keeps the entire vision of the problem at all stages.

To keep away from any kind of complexity, one works on a simplified case in which the master consists of a $2^h \times 2^h$ array of 2^{2h} base cells. Initially, the master is subdivided into regular 4×4 array of equally shaped super base cells, each of which could be seen as a new master that contains an array of $2^{h-2}\times 2^{h-2}$ base cells. The placement problem is solved for 4×4 array with relaxed overlapping condition, i.e. each cell is assigned to one of these 16 representative base cells, and each of these super bases is represented by the base cell in its lower left corner. In the subsequent step, the same thing would be done with super base cells. The procedure goes on until each super base cell becomes identical to a base cell.

4.2 The graph theoretic formulation of the placement of electronic circuit problem

The graph associated to the placement problem $G=(V,E,w)$ is as follows, where V denotes the set of the nodes $V=\{x_{i,k}/k=1,\dots,m \; et \; i=1,\dots,n\}$. E denotes the set of the edge and w denotes the set of the weights. An edge is introduced between $x_{i,k}$ and $x_{j,k}$ if and only if $i\neq j$. The weights are the coefficients of the matrix P. Furthermore, any feasible solution of the placement problem is associated with a clique in G of cardinality n, and its objective function value is the total weight of this clique.

A stable set in G is a subset of V in which no two nodes are connected by an edge. A clique is a subset

of nodes such that every pair of nodes is connected by an edge.
Thus, the placement problem is equivalent to the following graph theoretic problem. "Find among all cliques of cardinality n *in* G one of minimum weight". To solve this problem approximately, Jünger and all [20] used the heuristic which avoids the simultaneous deletion of more than one node. We iteratively delete nodes until a clique of cardinality n is left. To this end, we assign weights to the nodes in G and to the maximum stable sets in G. In each step we determine a maximum stable set S with the highest weight and a node u of S with highest weight (among all nodes of S). Then the node u is deleted from G. These steps are repeated until every maximum stable set consists of only one node. These remaining nodes define a clique of maximum cardinality.

5 Continuous Hopfield networks for the PECP

The main purpose of this section is to apply the CHN to solve the PECP. To this end, we define an appropriate energy function that enables us to resolve the PECP by the approach of CHN. To be precise, the choice of the parameters of this function must ensure the feasibility of the CHN equilibrium points.

5.1 Energy function for PECP

Let's consider a matrix array of $m \times n$ neurons represents an assignment of the n cells to the m base cells. The $m \times n$ neurons are grouped into n groups of m neurons. Each group of m neurons is used to represent the position of the cell on the integrated circuit.
For example, if $B = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$ represents the set of the cells, and $D = \{b_1, b_2, b_3\}$ represents the set of the base cells, a possible assignment is showed in Fig.4.
The second group of three neurons corresponds to the cell c_2 and $[0\ 0\ 1]$ means that cell c_2 is assigned to the base cell b_3.

$$\begin{array}{c} \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_8 \end{array} \begin{array}{c} \begin{array}{ccc} b_1 & b_2 & b_3 \end{array} \\ \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

Fig.4: A matrix array of 8x3 neurons to represent an assignment of the 8 cells to the 3 base cells.

To solve the PECP via the CHN, we choose an energy function which includes the objective function $f(v)$; in addition, it does not only penalize the linear constraint $Rv = b$ with a quadratic term, but also a linear term. Furthermore, the non-extreme values for any component v_i are also penalized.
Taking into consideration all these criteria, we propose the following energy function for the Placement of the PECP:

$$E(v) = \frac{\alpha}{2} \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{j=1}^{n} \sum_{l=1}^{m} v_{i,k} p_{ik,jl} v_{jl} + \frac{\eta}{2} \sum_{j=1}^{n} [e_j(v)]^2 + \beta \sum_{j=1}^{n} e_j(v) + \xi \sum_{j=1}^{n} \sum_{l=1}^{m} v_{j,l}(1 - v_{j,l}) \qquad (6)$$

Where

$$e_i(v) = \sum_{k=1}^{m} v_{i,k} \quad \forall i = 1, \ldots, n .$$

The first term in the energy function represents the cost function; the second term, which specifies the blocking constraints that every plant must be located at exactly one site, is penalized by $\eta \in \Re$. Analogously, all the linear constraints are penalized by the same parameter β. Finally, the term, which forces the analogue neurons to take finally discrete values 0 or 1, is penalized by the parameter ξ.
A simple comparison between the equation (3) and (6) implies

$$\begin{cases} T_{ik,jl} = -\alpha p_{ik,jl} - \eta \delta_{ij} + 2\delta_{ij}\delta_{kl}\xi \\ I_{ik} = -\beta - \xi \end{cases} \qquad (7)$$

where

$$\delta_{ij} = \begin{cases} 1 & if\ i = j \\ 0 & else \end{cases}$$

Giving the data n, m, C, d *and* O the PECP can be solved via the CHN, once the parameters η, ξ, β *and* α are determined.

5.2 Parameter settings

According to the equations (7), the weights and thresholds associated with PECP depend on the parameters α, η, β and ξ. Thus, in order to solve the PECP via the CHN, a convenient setting of these parameters is necessary. In this part, basing on an appropriate decomposition of the set $H_C - H_F$ and on the partial derivatives of the energy function, we choose the parameters of the energy function that assures the feasibility of the equilibrium points.
The partial derivatives of the generalized energy function are given by

$$E_{ik}(v) = \frac{\partial E(v)}{\partial v_{i,k}} = \alpha \sum_{j=1}^{n} \sum_{l=1}^{m} p_{ik,jl} v_{j,l} + \eta e_i(v) + \beta + \xi(1 - 2v_{i,k}) \quad (8)$$

To simplify, we impose the following constraints to the parameters of the energy function:
• The constraint below is imposed to minimize the objective function:

$$\alpha \geq 0$$

• The following constraint is naturally imposed to penalize the family of linear constraints of the PECP:

$$\eta \geq 0$$

• The next constraint is necessary to avoid the stability of the interior points $v \in H_C - H_F$:

$$T_{ik,ik} = -\eta + 2\xi \geq 0$$

Given the family of constraints of PECP:

$$e_i(v) = 1 \qquad \forall i \in \{1, \ldots, n\}$$

The partition of $H_C - H_F$ is defined as

$$\bullet\ U_1 = \{v \in H_C / \exists i, e_i(v) < 1\} \cap \{v \in H / \forall j, e_j(v) \leq 1\}$$

In this case, one cell $i \in \{1, \ldots, n\}$ exists such that $v_{i,k} = 0, \quad \forall k = 1, \ldots, m$ (this cell is not assigned to any base cell), thus the value $v_{i,k}$ will increase if $E_{i,k}(v) \leq -\varepsilon$.
Now the following constraint is obtained:

$$E_{ik}(v) \leq \alpha p_{\max} n + \beta + \xi \leq -\varepsilon$$
where $p_{\max} = \underset{\substack{i,j=1,\ldots,n \\ k,l=1,\ldots,m}}{Max} \{P_{ik,jl}\}$

The above constraint can be replaced by the following:

$$E_{ik}(v) \leq \alpha p_{\Sigma,\max} + \beta + \xi \leq -\varepsilon$$

where

$$p_{\Sigma,\max} = \underset{\substack{i=1,\ldots,n \\ l=1,\ldots,m}}{Max} \left\{ \sum_{j=1}^{n} \sum_{k=1}^{m} P_{ik,jl} \right\}$$

$$\bullet\ U_2 = \{v \in H_C\ / \exists i\ , e_i(v) > 1\}$$

In this case, one cell $i \in \{1, \ldots, n\}$ is assigned to two base cells $k \neq l$ so that $v_{i,k} = v_{i,l} = 1$ and Therefore the value $v_{i,k}$ will decrease if $E_{i,k}(v) \geq \varepsilon$.
The following constraint is obtained:

$$E_{ik}(v) \geq \alpha p_{\min} + 2\eta + \beta - \xi \geq \varepsilon$$

where

$$p_{\min} = \underset{\substack{i,j=1,\ldots,n \\ k,l=1,\ldots,m}}{Min} \{P_{ik,jl}\}.$$

Joining all these parametric constraints yields the following:

• Initial constraints:

$$T_{ik,ik} = -\eta + 2\xi \geq 0$$
$$\alpha \geq 0$$
$$\eta \geq 0$$

• U_1

$$\alpha p_{\max} n + \beta + \xi \leq -\varepsilon$$

where

$$p_{\max} = \underset{\substack{i,j=1,\ldots,n \\ k,l=1,\ldots,m}}{Max} \{P_{ik,jl}\}$$

or

$$\alpha p_{\Sigma,\max} n + \beta + \xi \le -\varepsilon$$

where

$$p_{\Sigma,\max} = \underset{\substack{i=1,\ldots,n \\ l=1,\ldots,m}}{Max} \left\{ \sum_{j=1}^{n} \sum_{k=1}^{m} P_{ik,jl} \right\}$$

• U_2

$$\alpha p_{\min} + 2\eta + \beta - \xi \ge \varepsilon$$

where

$$p_{\min} = \underset{\substack{i,j=1,\ldots,n \\ k,l=1,\ldots,m}}{Min} \left\{ P_{ik,jl} \right\}$$

Moreover, the solution can be feasible by choosing:

$$\xi = \frac{\alpha (np_{\max} - p_{\min}) + 2\varepsilon}{2} \qquad (9)$$

$$\eta = 2\xi \qquad (10)$$

$$\beta = -\frac{4\xi + \alpha (np_{\max} + p_{\min})}{2} \qquad (11)$$

This parameters depend on the parameters ε, α, m the number of base cells, n number of cells and on the matrix P.

6 Computational experiments

The Euler method proved to be highly sensitive, with respect to the initial conditions, and it requires a lot of CPU time for medium or greater size CHN instances. To overcome these shortcomes, we use the algorithm described in [28] to simulate the mapped PECP. This variable time-step method assures the feasibility of the equilibrium point and needs less CPU time. Besides, it is robust with respect to initial conditions.

The initial states are randomly generated:

$$v_{i,k} = 0.55 + 10^{-12} x, \quad x \in u[-0.4, 0.4]$$

where x is a random uniform variable in the interval $[-0.4, 0.4]$.

We choose the parameters

$\varepsilon = 10^{-6}$

$\alpha = \frac{1}{n \times m}$

ξ, η and β were computed from the equations (9), (10), (11).

We have performed experiments on three test circuits with $m = 16$ which are randomly generated. Circuit 1 consists of 1022 cells and its density is 50%. Circuit 2 consists of 2293 cells and its density is 23.6%. Circuit 3 consists of 2670 cells and its density is 50.3%. In order to compare the computational results obtained via the CHN with those obtained with the decomposition method, the training results reported, respectively, in table 1 and table 2 were performed on compatible IBM, Pentium(R) Dual-Core 2.5 GHz, 2.5 GHz, and 1 Go of RAM through Borland C++ Builder 3.

The CHN method performs much better than the decomposition method with respect to the mean estimated wiring length, and the CPU time.

Unlike the decomposition method, all CHN simulations yield valid placements.

Table 1 and table 2 list, respectively, the number of cells, the density of circuit, the number of simulations, the mean estimated wiring length and the mean of CPU time.

Table 1
Computational results of the placement of electronic circuits instances by decomposition method

number cells n	density of circuit%	Number of run	mean estimated wiring length	mean time (s)
1022	50	20	132309.5	5907.7
2293	23.6	15	722019.9	32881.2
2670	50.3	10	394361.78	44350.5

Table 2
Computational results of the placement of electronic circuits instances by CHN method

number cells n	density of circuit%	Number of run	mean estimated wiring length	mean time (s)
1022	50	20	54132.5	945.7
2293	23.6	15	77473.8	1719.1
2670	50.3	10	93762.13	1900.4

The final solution can be significantly improved by applying one of the following techniques. In the first technique, the parameter u_0 is not fixed but is gradually decreasing in time during the optimization process. As it reaches a critical value u_0^c, some of

the output variables $v_{i,k}$ begin to move significantly towards 0 or 1. The second technique, which can be combined with the first one described, involves adding random terms to the differential equation which characterizes the dynamics of the CHN; see [6].

7 Summary and conclusions

In this work, we have applied the CHN to solve the PECP. The PECP has been presented as a QKP. The resolution is based on an appropriate energy function; moreover, the variable time-step method, which replaces the Euler method, was used to solve the differential equation system associated to the CHN.

Additionally, and in order to solve the PECP, the mapping procedure and an appropriate parameter-setting procedure are discussed in detail. This procedure ensures that the solutions obtained are always feasible. The achieved results were compared with those attained through the decomposition method. The CHN method, unlike the decomposition method, takes less CPU time for medium or greater size PECP instances. Furthermore, the length of wiring used by the electronic circuits constructed via the decomposition method is longer than the one utilized with the CHN. Finally, all the attained assignments through the CHN are feasible, but with the decomposition approach the feasibility is not always guaranteed.

To make this approach more efficient, it can be combined with some metaheuristics, such as genetic algorithms [1] and [30], Tabu research, and Ant colony system, or it can be computationally optimized by introducing analytical improvements, such as replacing the hyperbolic tangent with a linear function.

Acknowledgements

The authors would like to express their sincere thanks for the referee for her/his helpful suggestions.

References:

[1] T. Bouktir, L. Slimani, Optimal power flow of the Algerian Electrical Network using genetic algorithms, *WSEAS*, Vol.3, Issue 6, pp. 1478-1483, 2004.

[2] D. Boyalakuntla, J. Murthy, Hierarchical compact models for simulation of electronic chip packages, *IEEE Transactions on Components and Packaging Technologies*, 25(4):629-34, 2002.

[3] R.D. Brandt, Y. Wang, A.J. Laub, S.K. Mitra, Alternative networks for solving the travelling salesman problem and the list-matching problem, *proceedings of the International Conference on Neural Networks*, vol. II, San Diego, USA, 1988, pp. 333-340.

[4] R. Burkard, Selected topics on assignment problems, *Technical Report 175, Techinshe Universitauml, t Graz, Austria*, November 1999.

[5] C. Chernlin, Implementation Feasibility of Convex Recursive Deletion Regions Using Multi-Layer Perceptrons, *WSEAS*, Volume 7, Issue 1, January 2008.

[6] A. Cichocki and R. Unbehauen, Neural Networks for Optimization and Signal Processing, B.*G. Teubner Stuttgart*, 1993.

[7] M. Ettaouil, Contribution à l'étude des problèmes de satisfaction de contraintes et à la programmation quadratiques en nombre entiers, allocation statiques de tâches dans les systemes distrubués, thèse d'état, *Université Sidi Mohammed ben Abdellah, F.S.T. de Fès*, 1999.

[8] M. Ettaouil, Y. Ghanou, Neural architectures optimization and genetic algorithms, *WESAS,* Volume 8, Issue 3, 526-537, March 2009.

[9] M. Ettaouil, K. Elmoutaouakil and Y. Ghanou, Artificial Neural Network and Placement of Electronic Circuits Problem, *HMAMS proceedings Universidad Rey Juan Carlos Madrid, Spain*, 2008.

[10] M. Ettaouil and C. Loqman, Constraint satisfaction problem solved by semi definite relaxation*, WESAS*, Volume 7,Issue 7, 951-961, 2008.

[11] Finke G, Burkard R, Rendl F. Quadratic assignment problems, *Annals of Discrete Mathematics*, 31:61-82, 1987.

[12] A. Frieze, L. Yadegar, On the quadratic assignment problem. *Discrete Applied Mathematics,* 5:89-98, 1983.

[13] M. R. Garey, D. S. Johnson , Computers and Intractability*, A Guide to the Theory of NP-completeness, Freeman, New York*, 1979.

[14] A.H. Gee, S.V.B. Aiyer, R.W. Prager, An analytical framework for optimizing neural networks, *Neural Networks*, 6, 79-97, 1993.

[15] A. Ghosh, S.K. Pal, Object Background classification using Hopfield type neural networks, *International Journal of Patten Recognition and artificial Intelligence*, 6, 989-1008, 1992.

[16] J.J. Hopfield, D.W. Tank, Neural computation of decisions in optimization problems, *Biological Cybernetics,* 52, 1-25, 1985.

[17] J.J. Hopfield, Neurons with graded response have collective computational properties like those of two-states neurons, *proceedings of the National academy of sciences of the USA,* 81, 3088-3092, 1984.

[18] V.R Joseph, Better Learning of Supervised Neural Networks Based on Functional Graph – An Experimental Approach, *WSEAS,* Volume 7, Issue 8, August 2008.

[19] M. Junger, V. Kaibel, Box-inequalities for quadratic assignment polytopes, *Mathematical Programming Series*, A 91:175-97, 2001.

[20] M. Jünger, A. Martin, G. Reinelt, R.Weismantel, Quadratic 0/1 Optimization and a Decomposition Approach for the Placement of Electronic Circuits, *Report 91. 102, Institut für Informatik Universität kôln*, 1991.

[21] R. M. Karp, Reducibility among combinatorial problems, *plenum Press*, New York, 1972.

[22] J. M. Kleinhans, G. Sigl, F.M. Joh annes, A new global optimization/rectangle dissection method for cell placement *, IEEE Int. Conference onCAD ICCAd-88*, pp. 506-509, 1988.

[23] L. Kaufman, F. Broeckx, An algorithm for the quadratic assignment problem using Benders decomposition, *European Journal of Operations Research,* 2:204-11, 1978.

[24] U. Lauther, A min-cut placement algorithm for general cell assemblies based on graph representation, *ACM/ICEE Proc*. 16th DAC, 1-10,1979.

[25] R. Lengauer, Combinatorial Algorithms for Integrated Circuit Layout, *Wiley-Teubner*, New-York, 1990.

[26] N. Queipo, G. Gil, Multiobjective optimal placement of convectively and conductively cooled electronic components on printed wiring boards, *Journal of Electronic Packaging*, 122(2), 152-9, 2000.

[27] N.M. Nasrabadi, C.Y. Choo, Hopfield network for stereo vision correspondence, *New York:Marcel Dekker*, 1994.

[28] P.M. Talaván, J. Yáñez, A continuous Hopfield network equilibrium points algorithm, *computers and operations research 32,* 2179-2196, 2005.

[29] K. Tatsumi, Y. Yagi & T. Tanino, (2002),Improved projection Hopfield network for the quadratic assignment problem, SICE 2002. *proceedings of the 41 st SICE annual conference,* vol. 4, pp.2295-2300, 2002.

[30] S.Yavuz, I. Cabar, O. Erol, Robot Mapping and Map Optimization Using Genetic Algorithms and Artificial Neural Networks, *WSEAS*, Issue 7, Volume 7, July 2008.