Mapping and Combining Combinatorial Problems into Energy Landscapes via Pseudo-Boolean Constraints

Conference Paper · October 2005

DOI: 10.1007/11565123_30 · Source: DBLP

4 authors:

Priscila M. V. Lima
Federal University of Rio de Janeiro

Glaucia K. E. Pereira
Imperial College London

Mariela Morveli-Espinoza
Federal University of Technology - Paraná/Brazil (UTFPR)

Felipe M.G. França
Federal University of Rio de Janeiro

Mapping and Combining Combinatorial Problems into Energy Landscapes via Pseudo-Boolean Constraints

Priscila M.V. Lima[1], Glaucia C. Pereira[2],
M. Mariela M. Morveli-Espinoza[2], and Felipe M.G. França[2]

[1] NCE/Instituto de Matemática, UFRJ, Rio de Janeiro, Brazil
priscila@nce.ufrj.br
http://www.geti.dcc.ufrj.br/
[2] COPPE – Sistemas e Computação, UFRJ, Rio de Janeiro, Brazil
{gpereira, mme, felipe}@cos.ufrj.br
http://www.cos.ufrj.br/∼felipe

Abstract. This paper introduces a novel approach to the specification of hard combinatorial problems as pseudo-Boolean constraints. It is shown (i) how this set of constraints defines an energy landscape representing the space state of solutions of the target problem, and (ii) how easy is to combine different problems into new ones mostly via the union of the corresponding constraints. Graph colouring and Traveling Salesperson Problem (TSP) were chosen as the basic problems from which new combinations were investigated. Higher-order Hopfield networks of stochastic neurons were adopted as search engines in order to solve the mapped problems.

Keywords: Higher-order Networks; Graph Colouring; Pseudo-Boolean Constraints; Satisfiability; Simulated Annealing; TSP.

1 Introduction

The ability to learn associative behaviour through examples is a desirable feature in an adaptive system. Nevertheless, it would not be practical to acquire, through examples, certain pieces of knowledge that had already been learnt by other systems. Besides, sometimes it is easier to describe a problem via its constraints to an artificial neural network (ANN) such that the set of its global energy minima corresponds to the set of solutions to the problem in question. For example, an explanation of how the Traveling Salesperson Problem (TSP) can be defined as a set of mathematical constraints that are solvable by an ANN can be found in [6] and [5].

Alternatively, constraints may be essentially logical, constituting a kind of description or specification of a suitable solution for a problem being modeled. A problem that apparently does not involve optimizing a cost function is that of finding a *model* for a logical sentence. In propositional logic, that would consist

M. De Gregorio et al. (Eds.): BVAI 2005, LNCS 3704, pp. 308–317, 2005.

of the assertion of truth-values to the propositional symbols that appear in the formula in question, in such a way that the formula as a whole becomes true. That mapping of truth-values to propositions constitutes, for propositional formulae, an interpretation of it [9]. A formula that has no models is said to be *unsatisfiable* or *inconsistent*. Some problems may be better described as a combination of logical and mathematical constraints. A subset of this combination could be seen as a sum of weighted products of boolean variables, pseudo-Boolean constraints [2].

This paper introduces a novel approach to the specification of hard combinatorial problems as pseudo-Boolean constraints defining an energy landscape representing the space state of solutions of the target problem. It is shown how easy is to map and combine different problems into new ones mostly via the union of the corresponding constraints. Graph colouring and Traveling Salesperson Problem (TSP) were chosen as the basic problems from which new combinations were investigated. Among other possible computational intelligence models that could have been used, (e.g., genetic algorithms, artificial immune systems, etc) this work adopted higher-order Hopfield networks of stochastic neurons in order to solve all the mapped problems.

2 Higher-Order Hopfield Networks

A notable step towards understanding the collective properties of artificial neural networks (ANNs) was taken by J. Hopfield [4] when he saw an analogy between the evolution of a spin-glass system towards minimizing its energy function and the evolution of the activity function of a so-called *Hopfield network*. For a function to be called an *energy function* it is necessary that its value decreases monotonically until the (or one of the) stable state(s) of the system is reached. The direct consequence of such interpretation is the proof of *convergence to energy minima* of artificial neural networks (ANNs) composed of symmetrically connected (i. e., $w_{ij} = w_{ji}$) McCulloch-Pitts' neurons $(i, j, \ldots)$ acting as energy minimization (EM) systems. The proof required the observation of a constraint: that nodes operate *asynchronously*, i.e., that no two nodes operate at the same time step. This restriction can be weakened to one where asynchronous operation is only required for *neighbouring nodes*, i.e., it is guaranteed that non-neighbouring nodes can operate at the same time and energy will still decrease monotonically [1]. Two nodes i and j are said to be *neighbours* if they are linked by a connection with weight $w_{ij} \neq 0$.

Sometimes it is convenient to express not only the mutual influence between two neurons, but also the influence of concurrent activation of three or more neurons. Such connections are known as multiplicative or *higher-order* and the number of units pertaining to a connection is called the *arity* of the connection. Only one value (positive or negative) is associated to each higher-order connection and networks containing one or more multiplicative connections are called *higher-order networks*. Notice that higher-order connections are still considered symmetric, i.e., they take part in the activation function of all nodes involved in the connection, and have the same weight value.

Unfortunately, a Hopfield network, even of higher-order, is only capable of finding local minima. In this sense, an improvement consists of incorporating an stochastic component to the neurons behavior such that the resulting network could find global minima through a mechanism known as *simulated annealing* [7]. In this way, consider a random variable d_i associated to each binary node $v_i \in V$, V denoting the set of random variables $v_1, v_2, \ldots, v_n$, $n = \mid V \mid$. The values of these random variables are taken from a common finite domain $D = \{0, 1\}$, so that v_i represents the state of neuron i and each element of D^n is a possible network state. Each $v_i \in V$ define a set of neighbours $Q(v_i)$ in such a way that a homogenous neighbourhood is obtained, i.e., for any two $v_i, v_j \in V$, if $v_j \in Q(v_i)$, then $v_i \in Q(v_j)$. The result of this incorporation can be described by the following equations:

$$\begin{cases} p(v_i = 1 | v_j = d_j; v_j \in Q(v_i)) = \frac{1}{1+e^{(-net_i)/T}} \\ p(v_i = 0 | v_j = d_j; v_j \in Q(v_i)) = \frac{e^{(-net_i)/T}}{1+e^{-net_i/T}} \end{cases}$$

Where $net_i = (\sum w_{ij} v_j(t)) - \theta_i$, θ_i is the threshold of neuron i, and T is the parameter known as *temperature* ($T \geq 0$).

3 Mapping Satisfiability to Energy Minimization

In order to convert satisfiability (SAT) to energy minimization (EM), consider the following mapping of logical formulae to the set $\{0, 1\}$:

$H(true) = 1$
$H(false) = 0$
$H(\neg p) = 1 - H(p)$
$H(p \wedge q) = H(p) \times H(q)$
$H(p \vee q) = H(p) + H(q) - H(p \wedge q)$

If a logical formula is converted to an equivalent in clausal form, the result being a conjunction φ of disjunctions φ_i, it is possible to associate energy to $H(\neg\varphi)$. Nevertheless, energy calculated in this way would only have two possible values: *one*, meaning solution not found (if the network has not reached global minimum), and *zero* when a model has been found. Intuitively, it would be better to have more “clues”, or degrees of “non-satisfiabililty”, on whether the network is close to a solution or not.

Let $\varphi = \wedge_i \varphi_i$ where $\varphi_i = \vee_j p_{ij}$, and p_{ij} is a literal. Therefore $\varphi = \vee_i \varphi_i$ where $\varphi_i = \wedge_j \neg p_{ij}$. Instead of making $E = H(\neg\varphi)$, consider $E = H^*(\neg\varphi) = \sum_i H(\neg\varphi_i)$. So, $E = \sum_i H(\wedge_j \neg p_{ij}) = \sum_i \prod_j H(\neg p_{ij})$, where $H(p)$ will be referred to as p. Informally, E counts the number of clauses that are *not satisfied* by the interpretation represented by the network's state.

An issue to point out is that the resulting network of the above mapping may have higher-order connections, i.e., connections involving more than two neurons. That does not constitute a hindrance as has been demonstrated that, with higher-order connections, Boltzmann Machines still converge to energy minima [3]. Remarks on a learning mechanics for this network are made in [6]. Parallel and distributed simulation of network with higher-order connections can be done by substituting each higher-order connection by a completely-connected subgraph. Alternatively, [10] converts the higher-order network to a binarily connected one that preserves the order of energy values of the different network states. A simple example demonstrates how SAT can be mapped to EM. Let φ be the formula, expressed as a conjunction of clauses:

$$\varphi = (p \vee \neg q) \wedge (p \vee \neg r) \wedge (r).$$

SAT(φ) can be translated to the minimum of the following energy function:

$$\begin{aligned} E &= H(\neg(p \vee \neg q)) + H(\neg(p \vee \neg r)) + H(\neg r) \\ &= H(\neg p \wedge q) + H(\neg p \wedge r) + H(\neg r) \\ &= (1-p) * q + (1-p) * r + (1-r) = q - pq - pr + 1 \end{aligned}$$

where $H(prop) = prop$.

4 Combinatorial Problems as Pseudo-Boolean Constraints

So far, the problem of mapping SAT to EM, by associating energy to "amount of non-satisfiability" and minimizing it, has been presented. This, together with the fact that the language of logic can be used to define a set of constraints, may lead to a technique for mapping and combining optimization problems into energy minimization. The mapping of three problems into constraint satisfiability are introduced next: TSP, Graph Colouring and a third problem resulting from the combination of the first two problems.

4.1 Mapping TSP

Let $G = (V, A)$ be an undirected graph, where V is the graph's vertex set, A the set of G's edges, being each edge an unordered pair of G's vertices. Associating each vertex $i \in V$ to a city and each edge $(i, j) \in A$ to a path between i and j, if $|V| = n \geq 3$ and $dist_{ij}$ is the cost associated to the edge $(i, j) \in A$ where $\{i, j\} \in V$, then, the Travelling Salesperson Problem (TSP) consists on determining the minimum cost Hamiltonian cycle of G. In order to enable the tour to end at an initial city a, a twin name a' is given so that it will be clamped as the least city of the tour (with all traveling costs repeated), in the same way that a is clamped as the first city of the tour. In this way, a problem with m cities has to use an augmented $n \times n$ matrix, where $n = m + 1$, so that all conditions may be applied to a round tour.

Mapping to Constraint Satisfiability. The network is composed by an $n \times n$ matrix of binary neurons v_{ij}, where i represents a city in V and j represents the position of i in the tour. The repetition of propositional clauses, which differ only by the value of indices, is represented in a compact form by the symbol of universal quantification. However, it should be stressed that the use of universal quantifiers to compress the representation of the propositional constraints does not mean that the language of logic used to describe such constraints has become first order logic. The network's behavior is specified by the following constraints:

Integrity Constraints:

(i) All n cities must take part in the tour:
$\forall i, \forall j | 1 \leq i \leq n, 1 \leq j \leq n : \vee_j (v_{ij})$. So, let $\varphi_1 = \wedge_i (\vee_j (v_{ij}))$.

(ii) Two cities cannot occupy the same position in the tour:
$\forall i, \forall j, \forall i' | 1 \leq i \leq n, 1 \leq j \leq n, 1 \leq i' \leq n, i \neq i' : \neg(v_{ij} \wedge v_{i'j})$.
So, let $\varphi_2 = \wedge_i \wedge_{i' \neq i} \wedge_j \neg(v_{ij} \wedge v_{i'j})$.

(iii) A city cannot occupy more than one position in the tour:
$\forall i, \forall j, \forall j' | 1 \leq i \leq n, 1 \leq j \leq n, 1 \leq j' \leq n, j \neq j' : \neg(v_{ij} \wedge v_{ij'})$.
So, let $\varphi_3 = \wedge_i \wedge_j \wedge_{j' \neq j} \neg(v_{ij} \wedge v_{ij'})$.

Optimality Constraints:

(iv) The cost between two consecutive cities in the tour:
$\forall i, \forall j, \forall i' | 1 \leq i \leq n, 1 \leq j \leq n-1, 1 \leq i' \leq n, i \neq i' : dist_{ii'}(v_{ij} \wedge v_{i'(j+1)})$
So, let $\varphi_4 = \vee_i \vee_{i' \neq i} \vee_{j<n} dist_{ii'}(v_{ij} \wedge v_{i'(j+1)})$.

Constraints **(ii)** and **(iii)** are Winner-Takes-All (WTA) constraints. They can be used to justify the conversion of disjunctions in the middle of constraints to a conjunction of disjuncts. All the constraints above are associated to a penalty strength that is expressed through multiplicative constants. The highest multiplicative constant, represented by β, is applied to the WTA constraints. The other integrity constraints (type **(i)**) are weighetd by α. The lowest penalty strength is given to optimality constraints (type **(iv)**), which are weighted by constant 1. So,

$$\begin{cases} dist = \max\{dist_{ij}\} \\ \alpha \quad = ((n^3 - 2n^2 + n) * dist) + h \\ \beta \quad = ((n^2 + 1) * \alpha) + h \end{cases}$$

Mapping SAT into EM. We will use the method described in [10] to map logical propositional formulae into the set $\{0, 1\}$. The H operator will be employed in all three problems approached by this work. The energy equation relative to the integrity constraints is presented next followed by the detailing of its components:

$E_i = \alpha H^*_{WTA}(\neg\varphi_1) + \beta H^*(\neg\varphi_2) + \beta H^*(\neg\varphi_3)$

As $\varphi_1 = \wedge_i(\vee_j(v_{ij}))$, $\neg\varphi_1 = \vee_i(\wedge_j(\neg v_{ij}))$.

$H^*(\neg\varphi_1) = \sum_{i=1}^{n} H(\wedge_j(\neg v_{ij})) = \sum_{i=1}^{n} \prod_{j=1}^{n} H(\neg v_{ij}) = \sum_{i=1}^{n} \prod_{j=1}^{n}(1 - v_{ij})$

However, due to WTA constraints, the actual mapping of $\neg\varphi_1$ is

$H^*_{WTA}(\neg\varphi_1) = \sum_{i=1}^{n} \sum_{j=1}^{n}(1 - v_{ij})$

As $\neg\varphi_2 = \vee_i \vee_{i' \neq i} \vee_j (v_{ij} \wedge_j v_{i'j})$,

$H^*(\neg\varphi_2) = \sum_{i=1}^{n} \sum_{i'=1, i' \neq i}^{n} \sum_{j=1}^{n} H(v_{ij} \wedge v_{i'j}) = \sum_{i=1}^{n} \sum_{i'=1, i' \neq i}^{n} \sum_{j=1}^{n} v_{ij} v_{i'j}$

As $\neg\varphi_3 = \vee_i \vee_j \vee_{j' \neq j} (v_{ij} \wedge_j v_{ij'})$,

$H^*(\neg\varphi_3) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{j'=1, j' \neq j}^{n} H(v_{ij} \wedge v_{ij'}) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{j'=1, j' \neq j}^{n} v_{ij} v_{ij'}$

Next, the term of the energy equation relative to the tour's cost (optimality constraints) is introduced: $E_o = \sum_s H^*(\varphi_4)$.

$H^*(\varphi_4) = \sum_{i=1}^{n} \sum_{i'=1, i' \neq i}^{n} \sum_{j=1}^{n-1} dist_{ii'} H(v_{ij} \wedge v_{i'(j+1)}) =$
$= \sum_{i=1}^{n} \sum_{i'=1, i' \neq i}^{n} \sum_{j=1}^{n-1} dist_{ii'} v_{ij} v_{i'(j+1)}$

The complete energy equation becomes: $E = E_i + E_o$.

4.2 Graph Colouring Mapping

Let $G = (V, A)$ be an undirected graph, where V is the graph's vertex set, A the set of G's edges, being each edge an unordered pair of G's vertices. The Graph Colouring Problem consists in determining the minimum assignment of colours (positive integers) to the vertices such that each vertex has only one colour and no two neighbouring vertices have the same colour.

Mapping to Constraint Satisfiability. The network is mainly composed by a matrix V_{colour} having $n \times n$ binary neurons vc_{ik} and a matrix C_{olour} having $1 \times n$ binary neurons c_k, where i is a vertex in V and k represents the colour associated to vertex i. Addicionally, a matrix $neigh_{ii'}$ is used to indicate the neighbouring relationship between vertices:

Integrity Constraints:

(v) Every vertex must have one colour assigned to it:
$\forall i, \forall_k | 1 \leq i \leq n, 1 \leq k \leq n : \vee(vc_{ik})$. So, let $\varphi_5 = \wedge_i(\vee_k vc_{ik})$.

(vi) Two neighbouring vertices cannot have the same colour:
$\forall i, \forall i', \forall_k | 1 \leq i \leq n, 1 \leq i' \leq n, 1 \leq k \leq n, i \neq i' :$
$\neg(neigh_{ii'}) \vee \neg(vc_{ik} \wedge vc_{i'k})$.
So, let $\varphi_6 = \wedge_i \wedge_{i' \neq i} \wedge_k (\neg(neigh_{ii'}) \vee \neg(vc_{ik} \wedge vc_{i'k}))$.

(vii) A vertex cannot have more than one colour:
$\forall i, \forall_k, \forall_{k'} | 1 \leq i \leq n, 1 \leq k \leq n, 1 \leq k' \leq n, k \neq k' : \neg(vc_{ik} \wedge vc_{ik'})$.
So, let $\varphi_7 = \wedge_i \wedge_k \wedge_{k' \neq k} \neg(vc_{ik} \wedge vc_{ik'})$.

(viii) If a colour k is assigned to a vertex in matrix V_{colour}, then the corresponding unit in matrix C_{olour} must be activated:
$\forall i, \forall k | 1 \leq i \leq n, 1 \leq k \leq n : \neg vc_{ik} \vee c_k$. So, let $\varphi_8 = \wedge_i \wedge_k (\neg vc_{ik} \vee c_k)$.

Optimality Constraints:

(ix) The number of activated elements in matrix C_{olour}:
$\forall k | 1 \leq k \leq n : c_k$. So, let $\varphi_9 = \vee_k c_k$.

Similarly to the case of TSP, multiplicative constants α and β are used to indicate the penalty strength:

$$\begin{cases} \alpha = (n * 1) + h \\ \beta = ((n^3 + n^2 + 1) * \alpha) + h \end{cases}$$

Mapping SAT into EM. Let's generate the energy equation relative to the integrity constraints: $E_i = \beta[H^*(\neg\varphi_7)] + \alpha[H^*_{WTA}(\neg\varphi_5) + H^*(\neg\varphi_6) + H^*(\neg\varphi_8)]$. Since $E_o = \sum_s H^*(\varphi_9)$, then

$$E = E_i + E_o = \beta[\sum_{i=1}^n \sum_{k=1}^n \sum_{k'=1, k' \neq k}^n vc_{ik} vc_{ik'}] + \alpha[\sum_{i=1}^n \sum_{k=1}^n (1 - vc_{ik})] + \alpha[\sum_{i=1}^n \sum_{i'=1, i' \neq i}^n \sum_{k=1}^n vc_{ik} vc_{i'k} neigh_{ii'}] + \alpha[\sum_{i=1}^n \sum_{k=1}^n vc_{ik}(1 - c_k)] + \sum_{k=1}^n c_k$$

4.3 Map Colouring-TSP Mapping

A combination of two different problems is tackled here: Map Colouring and TSP. This hybrid problem is based on a set of cities, which are organised in contiguous regions. The TSP restrictions are maintained and the neighbourhood among adjacent regions is represented by different colours. The cost functions of the original problems, i.e., number of colours and tour cost, are part of the new cost function to be minimized. Interesting solutions would be tradeoffs between solutions of the two problems and this could be obtained by minimizing the change of colours between consecutive cities in the tour.

Let $M = (V, A_1, A_2)$ be an undirected multigraph, where V is the graph's vertex set, being each vertex $i \in V$ associated to a city. A_1 is the set of M's edges so that an edge $(i, j) \in A_1$ exists iff i and j belong to different adjacent regions. A_2 is the set of M's edges associated to all possible direct paths between any pair of cities i and j. Each edge $(i, j) \in A_2$ has an associated distance cost $dist_{ij}$. The resulting Map Colouring-Travelling Salesperson Problem (MC-TSP) consists of determining (i) a tour and (ii) a colour assignment to the different regions (by assigning colours to the visited cities).

Mapping to Constraint Satisfiability. The resulting network is composed by the matrices devised for (a) Graph Colouring and (b) TSP:

(a) A matrix V_{colour} having $n \times n$ binary neurons vc_{ik} and a matrix C_{olour} having $1 \times n$ binary neurons c_k, where i is a vertex in V and k represents the colour associated to vertex i. Addicionally, an $n \times n$ matrix $neigh$ is used to indicate the neighbouring relationship between vertices;
(b) An $n \times n$ matrix of binary neurons v_{ij},
where i represents a city in V and j represents the position of i in the tour.

Integrity Constraints:

The set of integrity constraints is the union of TSP's integrity constraints **(i)**, **(ii)**, **(iii)** and graph colouring's integrity constraints **(v)**, **(vi)**, **(vii)**, **(viii)**.

Optimality Constraints:

The set of optimality constraints is the union of TSP's and Graph Colouring's optimality constraints **(iv)**, **(ix)** and constraints of type **(x)** below:

(x) The change of colours between consecutive cities in the tour:
$\forall i, \forall j, \forall i', \forall k, \forall k' | 1 \leq i \leq n, 1 \leq j \leq (n-1), 1 \leq i' \leq n, 1 \leq k \leq n, 1 \leq k' \leq n, i \neq i', k \neq k' : (v_{ij} \wedge v_{i'(j+1)} \wedge vc_{ik} \wedge vc_{i'k'})$.
So, let $\varphi_{10} = \vee_i \vee_{j<n} \vee_{i' \neq i} \vee_k \vee_{k' \neq k} (v_{ij} \wedge v_{i'(j+1)} \wedge vc_{ik} \wedge vc_{i'k'})$.

Multiplicative constants γ and δ are added to the multiplicative constants of TSP and Graph Colouring in order to indicate the new penalty strengths:

$$\begin{cases} dist = \max\{dist_{ij}\} \\ \alpha \quad = ((n^3 - 2n^2 + n) * dist) + h \\ \beta \quad = ((n^5 - n^4 - n^3 + n^2 + 1) * \alpha) + h \\ \gamma \quad = ((n+1) * \beta) + h \\ \delta \quad = ((2n^3 - n^2 + n + 1) * \gamma) + h \end{cases}$$

Mapping SAT into EM. The energy equation relative to the integrity and optimality constraints are:

$E_i = \delta[H^*(\neg\varphi_2) + H^*(\neg\varphi_3) + H^*(\neg\varphi_7)] + \gamma[H^*_{WTA}(\neg\varphi_1) + H^*_{WTA}(\neg\varphi_5) + H^*(\neg\varphi_6) + H^*(\neg\varphi_8)]$, and

$E_o = \beta[H^*(\varphi_9)] + \alpha[H^*(\varphi_{10})] + H^*(\varphi_4)$.

Finally, $E = E_i + E_o$. Notice that E_o above corresponds to a possible way of combining the two original problems. In this case, minimizing the number of colours has been prioritized over the other two components of E_o, namely φ_{10} and φ_4. Similarly, φ_{10} has been prioritized over φ_4. Different priority orders could be explored originating the specification of new problems. In fact, the possibility of combining a multitude of problems,is quite an interesting feature

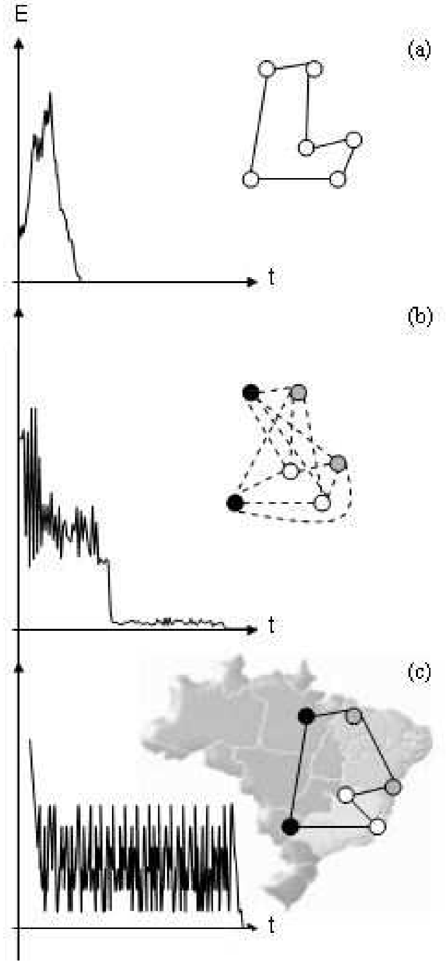

Fig. 1. Samples of the energy (E) behaviour and global minima found in (a) TSP, after 560 steps (t); (b) Graph Colouring, after 9967 steps (t), and (c) Map Colouring–TSP, after 38277 steps (t). Geometrical cooling (0.99) was used in (a), (b) and (c).

of our modeling, since real practical problems requiring optimization treatment are often not reducible to a single combinatorial problem. Figure 1 illustrates experimental results from the mapping of the three problems over simple six nodes graphs into stochastic high-order networks.

5 Conclusion

Although there are already language proposals oriented to the specification of problems via sets of constraints, e.g., Z notation [11], the possibility of combining different sets of such constraints in order to specify a new target problem is the main contribution of this work. Moreover, our approach profits from the intermediate definition of an energy function, which can be minimized by any available solver, not only higher-order Hopfield networks of stochastic neurons, as considered in this work. The development of a compiler which translates constraints into high-order networks and the mapping of molecular modeling via pseudo-boolean constraints are ongoing work. Among the most interesting investigations for future work, we intend to develop an integration of first-order logic inferencing [8] with pseudo-boolean constraints as an alternative and natural way of processing constraint logic programming.

References

1. Barbosa, V.C, Lima P.M.V.: On the distributed parallel simulation of Hopfield's neural networks. Software-Practice and Experience **20**(10) (1990) 967–983.
2. Dixon, H.E., Ginsberg, M.L., Parkes, A.J.: Generalizing Boolean Satisfiability I: Background and Survey of Existing Work. Journal of Artificial Intelligence Research **21** (2004) 193–243.
3. Geman S., Geman D. : Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6** (1984) 721–741.
4. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proc. of the National Academy of Sciences USA **79** (1982) 2554–2558.
5. Hopfield, J.J., Tank D.W.: Neural computation of decisions in optimization problems. Biological Cybernetics **52** (1985) 141–152.
6. Jones, A.J.: Models of Living Systems: Evolution and Neurology. Lecture Notes. Department of Computing. Imperial College of Science, Technology and Medicine, London, UK (1994).
7. Kirkpatrick, S., Gellat Jr., C.D., Vecchi, M.P.: Optimization via Simulated Annealing. Science **220** (1983) 671–680.
8. Lima P.M.V.: Resolution-Based Inference on Artificial Neural Networks. Ph.D. Thesis, Department of Computing. Imperial College of Science, Technology and Medicine, London, UK (2000).
9. Lima P.M.V.: A Goal-Driven Neural Propositional Interpreter. International Journal of Neural Systems **11** (2001) 311–322.
10. Pinkas, G.: Logical Inference in Symmetric Neural Networks. D.Sc. Thesis, Sever Institute of Technology, Washington University, Saint Louis, USA (1992).
11. Mike Spivey: The Z Notation: A Reference Manual. 2nd edition, Prentice Hall International Series in Computer Science (1992).