Appl. Math. and Comp. Sci., 1994, vol.4, No.3, 475–482

APPLICATION OF A BOLTZMANN MACHINE TO TIMETABLE DESIGN

KONRAD PIWAKOWSKI*, MAREK TROJANOWICZ*

This paper contains a description of the construction of a Boltzmann machine that can be applied to obtain a class-teacher timetable. In particular, this approach concerns requirements that are characteristic to primary and secondary schools. The conditions of asymptotic convergence to an optimal solution are presented.

1. Introduction

Constructing school timetables belongs to the classical discrete optimization problems. The earliest research on applying a computer to this problem was started in the beginning of the 1960s. Even *et al.* (1975) showed that in a general case this problem is NP-complete which implies that the automatic finding of the optimal solution is practically impossible even for small inputs. Recently, among many suboptimal methods applied to solving this problem one can distinguish heuristic techniques: genetics algorithms (Colorni *et al.*, 1992) and simulated annealing (Abramson, 1987), as the most promising. Application of a Boltzmann machine leads to highly distributed processing of the computation. This feature could be very helpful when dealing with the great computational complexity of the problem. The general definition of the timetable problem is as follows.

Let U, H, S denote the set of participants (i.e. teachers, classes, educational equipment), set of "hours" (time slots) and set of classrooms, respectively. Functions $h : U \rightarrow \mathbb{P}(H)$ and $s : U \rightarrow \mathbb{P}(S)$ define time and the set of classrooms in which particular participants are able to participate in any meet (because of the fact that some of the subjects have to be realized in specified classrooms, they can also be regarded as individual participants). Let L denote the set of meets and function $u : L \rightarrow \mathbb{P}(U)$ assign a set of participants to each meet. Constructing the timetable is based on finding a multiple-valued function $r : L \rightarrow H \times S$, $r(l) = (r_h(l), r_s(l))$ that fulfils a list of conditions, among which the following ones are the most important

a) $\underset{l \in L}{\forall}\ r(l) \in \bigcap_{u \in u(l)} h(u) \times s(u)$, i.e. time and place assigned to the lesson are suitable for each participant;

b) $\underset{l_1, l_2 \in L}{\forall}\ l_1 \neq l_2 \wedge r_h(l_1) = r_h(l_1) \Rightarrow u(l_1) \cap u(l_2) = \emptyset$, i.e. each participant participates in at most one lesson at the same time.

* Technical University of Gdańsk, Department of Foundations of Informatics
80–950 Gdańsk, ul. Narutowicza 11/12, Poland

In practice there are many other conditions such as avoiding gaps between lessons, regularity of arranging the number of busy hours in particular days from the point of view of some of the participants and many others.

The network considered below is defined in such a way that any of the temporary states can be interpreted as a certain relation defined in space $L \times H \times S$. The matter of implementation is in defining the rule of fixing the values of strengths of particular links with assigned functions h, s and u, taking into account conditions (a), (b) and, if necessary, others, in such a way that at the moment of reaching stability the state of network will, with great probability, be expressing function r fulfilling all the required conditions. In our construction we obtained the asymptotic convergence of the global network state to the optimal configuration and the correspondence between every stable state and relation r being a partial function, fulfilling conditions (a) and (b).

2. The Network Construction

In the sequel, the following notation is adopted:

- $y(i,t)$ denotes the state of the i-th unit at moment t,
- $y(t) = (y(1,t), y(2,t), \ldots, y(n,t))$ denotes the global state of the network at moment t,
- $y^*(t,i) = (y(1,t), y(2,t), \ldots, y(i-1,t), 1-y(i,t), y(i+1,t), \ldots, y(n,t))$ denotes the neighbouring state, in which only the state of the i-th unit is different from that one in state $y(t)$,
- $w(i,j)$ denotes the strength of connection of the output of unit j with the input of unit i,
- $E(y,t) = \sum\limits_{i,j} y(i,t)y(j,t)w(i,j)$.

By (sequential) Boltzmann machine we mean the neural network that fulfils the following conditions:

- $y(i,t) \in \{0,1\}$ for each unit i at each moment t,
- $w(i,j) = w(j,i)$ for each pair of units i,j.
- In every step t a random choice of one unit takes place, and then its state is being changed into the opposite one with probability

$$p = \frac{1}{1 + \exp\left(\dfrac{-\Delta(i,t)}{c(t)}\right)} \tag{1}$$

where i is the number of a chosen unit, $\Delta(i,t) = E(y^*(t,i)) - E(y(t))$, and $c : \mathbb{N} \to \mathbb{R}_+$ is a decreasing function.

In practice, the network specified in this way, for $t \to \infty$, $c(t) \to 0$, always reaches a stable state y, where the state function $E(y,t)$ reaches its local maximum and for $c(t) = c = \text{const}$, we have

$$\lim_{c \to 0} \lim_{t \to \infty} p\Big(y(t) \in Opt\Big) = 1 \tag{2}$$

where *Opt* denotes the set of these states y, in which $E(y)$ reaches the global maximum (Aarts and Korst, 1989).

Assume the following notation:

$K = \{k_1, k_2, ..., k_{k0}\}$ — set of classes,
$N = \{n_1, n_2, ..., n_{n0}\}$ — set of teachers,
$S = \{s_1, s_2, ..., s_{s0}\}$ — set of classrooms,
$D = \{d_1, d_2, ..., d_{d0}\}$ — set of weekdays,
$G = \{g_1, g_2, ..., g_{g0}\}$ — set of hours,
$L = \{l_1, l_2, ..., l_{l0}\}$ — set of lessons.

Assume that the number of hours g_0, in which meets can take place is the same every day. None of the weekdays nor hours are preferred. Duration of each hour is the same and hour $g_i + 1$ follows directly hour g_i. With every lesson $l \in L$ there is associated a set of participants $u(l) = \langle n(l), k(l), s^*(l), p(l)\rangle$, where $n(l) \in N$, $k(l) \in K$, $s^*(l) \subseteq S$, $p(l)$ is the name of a subject. Set $s^*(l)$ denotes a classroom where a given lesson can, but does not have to, take place and because of that it is not a participant accordingly to the definition presented in the introduction.

In this case, the crucial point of constructing the timetable is in finding the function $r : L \rightarrow S \times D \times G$ $(r(l) = (s(l), d(l), g(l)))$ which fulfils the following conditions:

$$\underset{l \neq l' \in L}{\forall} \; n(l) = n(l') \Rightarrow d(l) \neq d(l') \vee g(l) \neq g(l') \tag{3}$$

i.e. no teacher participates in two lessons at the same time,

$$\underset{l \neq l' \in L}{\forall} \; k(l) = k(l') \Rightarrow d(l) \neq d(l') \vee g(l) \neq g(l') \tag{4}$$

i.e. no class participates in two lessons at the same time,

$$\underset{l \neq l' \in L}{\forall} \; s(l) = s(l') \Rightarrow d(l) \neq d(l') \vee g(l) \neq g(l') \tag{5}$$

i.e. no classroom is used by two classes at the same time,

$$\underset{l \in L}{\forall} \; s(l) \in s^*(l) \tag{6}$$

i.e. lesson takes place in one of the permitted classrooms,

$$\underset{k \in K}{\forall} \; \underset{d,d' \in D}{\forall} \left|\left\{l \in L | k(l) = k \wedge d(l) = d\right\}\right| = \left|\left\{l \in L | k(l) = k \wedge d(l) = d'\right\}\right| \tag{7}$$

i.e. each class has the same number of lessons every day — for simplicity we assume that the number of days divides the sum of all the lessons of any class,

$$\forall_{k\in K}\ \forall_{d,\in D}\ \forall_{i,j\in \mathbb{N}}\ \exists_{l,l'\in L}\Big(k(l)=k(l')=k\wedge d(l)=d(l')=d\wedge g(l)=l_i\wedge g(l')=l_j\Big)$$

$$\Rightarrow\Big(\forall_{h\in \mathbb{N}}\ i<h<j\Rightarrow \exists_{l''\in L}\ k(l'')=k\wedge d(l'')=d\wedge g(l'')=g_h\Big) \tag{8}$$

i.e. classes do not have breaks between meets — i.e. (gaps).

In addition, function r should:

i) minimize the number of gaps between the meets of particular teachers,

ii) assign of the same subjects to the most distant days of the week for each class (It often happens in practice that lessons of the same subject are being joined together, but we omit such a case in our consideration — however, it could be regarded in our approach without any difficulties).

Now we are going to present the network that is responsible for finding the function r fulfilling conditions presented above. Each element $(l,s,d,g)\in L\times S\times D\times G$ such that $s\in s^*(l)$ corresponds to exactly one unit which will be denoted by $n_{l,s,d,g}$. The strength assigned to each connection is determined by the following formula:

$$w(n_{l,s,d,g},n_{l',s',d',g'})=\sum_{i=1}^{8}w_i(n_{l,s,d,g},n_{l',s',d',g'}) \tag{9}$$

where components w_i are denoted in the following Tab. 1:

The global state $y(t)$ of so denoted network determines relation $r\subseteq L\times S\times D\times G$ i.e. $(l,s,d,g)\in r$ iff $y(n_{l,s,d,g},t)=1$. Of course, we expect that after reaching stability of the network the relation r will be function $r:L\rightarrow S\times D\times G$ and will fulfil conditions (3)–(8) and (i), (ii). The analysis presented below shows partial meeting of these exceptions.

Property 1. *If parameters of the network $(a,b,...,f)$ satisfy the following conditions:*

$$a>0, e>0, c_{na}<-(a+3e) \tag{10}$$

$$b, c_{kl}, c_{na}, d<-(a+2e) \tag{11}$$

$$f\leq 0 \tag{12}$$

then every state y corresponding to a local maximum of the state function $E(y)$ determines a partial function $r:L\rightarrow S\times D\times G$ fulfilling conditions (3)–(6).

Proof. (Because of its technical character, we present only a sketch of the proof).

1) We are going to show that state y, where the time conflict of any teacher takes place, cannot correspond to the local maximum of function $E(y)$. We choose a teacher (n), day (d) and hour (g) so that the number of lessons m, in which teacher n takes part at the same time on day d at hour g, is maximal (for all triples: n, d, g), while g is the earliest hour that could have been chosen (when, in many cases, the time conflict of a teacher takes maximal value). We choose a random active unit $n_{l,s,d,g}$ such that $n(l)=n$ (at least two such units exist if the time conflict of

Tab. 1. Description of component strengths of links of network.

i	$w_i(n_{l,s,d,g}, n_{l',s',d',g'})$	Comments
1	$\begin{cases} a & \text{iff } l=l' \land s=s' \land g=g' \land d=d' \\ 0 & \text{otherwise} \end{cases}$	$a > 0$; each unit has small positive autoconnection. If any other unit, which is connected by negative connection, is active, then it becomes active itself.
2	$\begin{cases} b & \text{iff } l=l' \land (s=s' \lor g=g' \lor d=d') \\ 0 & \text{otherwise} \end{cases}$	$b < 0$; at most one unit corresponding to a given lesson should be active.
3	$\begin{cases} c_{kl} & \text{iff } k(l)=k(l') \land l \neq l' \land g=g' \land d=d' \\ 0 & \text{otherwise} \end{cases}$	$c_{kl} < 0$; connection tends towards avoiding time conflicts of particular classes (cf. (3)).
4	$\begin{cases} c_{na} & \text{iff } n(l)=n(l') \land l \neq l' \land g=g' \land d=d' \\ 0 & \text{otherwise} \end{cases}$	$c_{na} < 0$; connection tends towards avoiding time conflicts of particular teachers (cf. (4)).
5	$\begin{cases} c_{sa} & \text{iff } s(l)=s(l)' \land l \neq l' \land g=g' \land d=d' \\ 0 & \text{otherwise} \end{cases}$	$c_{sa} < 0$; connection tends towards avoiding time conflicts of particular classrooms (cf. (5)).
6	$\begin{cases} d & \text{iff } k(l)=k(l') \land l \neq l' \land d=d' \land g=g_i \\ & \land g'=g_j \land \lvert i-j \rvert \geq AvLes(k) \\ 0 & \text{otherwise} \end{cases}$	$d < 0$ $AvLes(k)$ denotes average number of lessons of the class k during one day. Such a connection leads to regularity of arranging meets in the week (cf. (7)) and to exclusion of gaps between meets of particular classes (cf. (8)).
7	$\begin{cases} e & \text{iff } n(l)=n(l') \land l \neq l' \land d=d' \land g=g_i \\ & \land g'=g_j \land \lvert i-j \rvert = 1 \\ 0 & \text{otherwise} \end{cases}$	$e > 0$; connection tends towards increasing the probability of constructing the timetable where teachers often have meets following directly each other (cf. (i)).
8	$\begin{cases} f(d,d') & \text{iff } k(l)=k(l)' \land l \neq l' \land p(l)=p(l') \\ 0 & \text{otherwise} \end{cases}$	$f(d,d') < 0$; function f denotes how much inadvisable it is for one class to have lessons of the same subject on days d and d'. The closer day d approaches (in time) d', the less value f is (cf. (ii)).

a teacher took place). Let us show that the change of state of such a unit will cause a positive increase of the state function $\Delta E > -\Big((m-1)c_{na} + a + (2m-1)e\Big) > 0$ (by the term of (10) and $m \geq 2$).

2) Based on 1) in every state y corresponding to the local maximum $E(y)$ the sum of positive stimulation of an active unit does not exceed $a + 2e$ (only a and e strengths are positive). Now it is easy to show the fulfilment of conditions (3)–(5) and that r has to be a partial function (i.e. each lesson corresponds at most to one active unit) on the basis of assumption (11).

3) Fulfilment of condition (6) is obvious. ■

Property 2. *If the parameters of network* $(a, b, \ldots, f)$ *satisfy conditions (10)–(12),* y *is the state of the network corresponding to a local maximum of the state function* $E(y)$ *and relation* r *assigned by state* y *is a total function* $r: L \to S \times D \times G$*, then* r *satisfies conditions (3)–(8).*

Proof. Fulfilment of conditions (3)–(6) results from Property 1. Fulfilment of conditions (7) and (8) due to the assumption that r is total function, is guaranteed by the component w_6 of the strengths of links. It causes exclusion of the situation where a certain class k participates in meets on the same day d at two different hours by g, g' "distant from each other" cf. not less than $AvLes(k)$ (Tab. 1).

Thus, each class k participates in at most $AvLes(k)$ lessons a day. Meeting the assumption that r is a total function, fulfilment of condition (4) and the assumption that the average number of lessons $AvLes(k)$ falling on one day for each class k is integer, we obtain that each class k participates precisely in $AvLes(k)$ lessons a day, which implies condition (7). Fulfilment of conditions (8) is a direct consequence of condition (7) and of the above considerations. ■

Property 3. *If parameters of the network* $(a, b, \ldots, e)$ *satisfy conditions (10), (11) and*

$$f = 0,\ a > e \tag{13}$$

then, if there exists a total function $r: L \to S \times D \times G$ *fulfilling conditions (3)–(8), every state of the network* y *corresponding to the global maximum of the state function* $E(y)$ *determines such a function* r.

Proof. We presented only the outline.

1) Based on Property 1 the relation determined by the state corresponding to the global maximum of the function $E(y)$ has to be a partial function fulfilling conditions (3)–(6).

2) If $f = 0$, the relation r, corresponding to the state of network, is a partial function and conditions (3)–(5) are fulfilled, then links of the negative strengths do not influence the value of the state function.

3) If $a > e$, condition (3) is fulfilled and links of the negative strengths do not influence the value of $E(y)$, then the state function takes on the greater value the more units are active. Of course, there can be no more than $|L|$ active units — otherwise negative links w_2 would have influenced the value of the state function.

4) Immediately from 3) it follows that r is a total function, and then, because of Property 2, conditions (7) and (8) are fulfilled. ■

Remark 1. Taking into account requirements (i) and (ii) this is realized by introducing component strengths of links w_7 and w_8. Establishing sufficiently small values of parameters $f(d, d')$ for proper pairs of days (d, d') is it possible to assert a required time distance between lessons of the same subjects for each class, in every state of the network corresponding to the local maximum of state function. Unfortunately, we cannot formulate a similar property for condition (i).

3. Concluding Remarks

Properties presented above lead to the conclusion that if parameters $a, b, ..., f$ are properly chosen, then every stable state of the network corresponds to the timetable without any time conflicts, but some of the lessons might be omitted. In that case, from the point of view of particular class there is no guarantee that there are no gaps between the meets or that there is an irregular distribution of meets in week. On the other hand, Property 3 and (2) guarantees the construction of a complete timetable satisfying conditions (7) and (8) with arbitrary great probability (provided that such a timetable exists) with a sufficiently small value $c(t) = c = \text{const}$ and time of networks functioning long enough. In practice, despite the asymptotic warrants, application of function $c(t) = c = \text{const}$ is inefficient. With great values c the network behaves completely chaotic, with small ones the stable state corresponding to the local maximum of the state function is being quickly determined and further changes leading possibly towards the global maximum occur too seldom. The crux of the Boltzmann machine is in application of the decreasing function $c(t) \to 0$, that gives the best practical result. The slower $c(t)$ decreases the longer time is needed to reach the equilibrium, however, the probability of constructing the timetable that fulfills all required conditions increases.

The neural network described above was implemented on a sequential computer (IBM PC 486). The largest input data were 12 classes, 16 teachers, 14 classrooms, with the total number of 300 lessons per week. Results of simulation confirmed the presented properties and showed the possibility of applying of this idea in practice. However, the satisfying efficiency for real (even small) data can be reached only by application of specified, concurrent hardware realizing the artificial neural networks.

Finally, we would like to list the most important, in our opinion, advantages of the approach:

- The construction is easy to modify. Therefore, additional requirements such as: time restrictions of particular teachers, grouping some of the meets, etc., can be met.
- Description of the method (i.e. list of units and connections) is declarative rather than algorithmic, so that experiments can be carried out not only by programmers.
- Because of the nature of the Boltzmann machine, computations can be carried out in a distributed environment.

References

Aarts E. and Korst J. (1989): *Simulated Annealing and Boltzmann Machines.* — New York: John Wiley & Sons.

Abramson D. (1987): *Constructing school timetables using simulated annealing; sequential and parallel algorithms.* — Technical Report 112069R, Royal Melbourne Institute of Technology.

Colorni A., Doriego M. and Maniezzo V. (1992): *On the use of the genetic algorithms to solve the time-table problem.* — Technical Report 90–060, Politecnico di Milano.

Even S., Itai A. and Shamir A. (1975): *On the complexity of timetable and multicommodity flow problems.* — Proc. 16th IEEE Annual Symp. *Foundations of Computer Science*, New York, pp.184–193, and SIAM J. Comput, v.5, No.??, pp.691–703.