Transaction Date:
5/14/2007 04:38:28 PM

WEB DELIVERY

ILLiad TN: 338397

Journal Title: 'Neurocomputing', NATO ASI Lecture Notes Series F , volume 68 = Neuro-computing : algorithms, architectures, and applications
Book Title:
Book Author:

Volume: Issue:
Month/Year: 1990 **Pages:** 51-62

Article Author: Robert Azencott
Article Title: Synchronous Boltzmann machines and Gibbs fields

Additional Information:

FP

Patron Information:

robert azencott
832 330 4706
razencot@math.uh.edu
1230 harvard street
Houston, TX 77008

Thank You for Using Our Services!

Contact Us: 713-743-9720 / ill@lib.uh.edu
Office Hours: 8am – 5pm, Mon - Fri

Recognition Networks. In
1975).
:w approach to learning in
cs: the Tiling Algorithm,
ıral Network for Pattern
ɔnal Conference on Neural
K.: Phoneme Recognition
ɔn Acoustics, Speech and
al Recognition. Snowbird
ıpril 1989).
cessing. Bradford Books,
n Wiley & Sons, (1973).
[A,(1969).
nal Conference On Pattern
J.Howard, R., Jackel, L.:
mplex Systems vol 1, 877-

SYNCHRONOUS BOLTZMANN MACHINES AND GIBBS FIELDS : LEARNING ALGORITHMS

Robert AZENCOTT (*)
École Normale Supérieure and Université Paris-Sud

INTRODUCTION

The Boltzmann machines are stochastic networks of formal neurons linked by a quadratic energy function. Hinton, Sejnowski and Ackley who introduced them as pattern classifiers that learn, have proposed a learning algorithm for the asynchronous machine. Here we study the synchronous machine where all neurons are simultaneously updated, we compute its equilibrium energy, and propose a synchronous learning algorithm based on *delayed* average coactivity of pairs of connected neurons. We generalize the Boltzmann machine paradigm to much wider types of interactions and energies allowing multiple interactions of arbitrary order. We propose a learning algorithm for these generalized machines using the theory of Gibbs fields and parameter estimation for such fields. We give quasi-convergence results for all these algorithms, within the framework of stochastic algorithms theory. The links between generalized Boltzmann machines and Markov field models sketched here provide the groundwork for designing generalized Boltzmann machines capable of performing efficient low level vision tasks. These Boltzmann vision modules are described in a forthcoming paper.

The asynchronous Boltzmann machines are widely considered as slow learners. However we emphasize here *their use at suitably selected fixed temperatures*, to avoid long stabilization times due to vanishing temperatures. Moreover *the synchronous versions studied here are structurally much faster*, if implemented on parallel hardware. In collaboration with P. Garda (I.E.F., Université Paris-Sud) and other researchers at DIAM (École Normale Supérieure, Paris), we are currently studying the technological feasibility of such specialized hardware implementations. We feel that Boltzmann machines, in suitable synchronous versions, and with general energies offer a world of learning networks with exciting technological capabilities as well as interesting tentative models for low level human vision.

(*) Chercheur associé aux lab. CNRS : [LMENS] et [Stat. Appliquées Univ. Paris-Sud].
Address : ENS, 45 rue d'Ulm - F-75230 PARIS CEDEX 05 (France)

NATO ASI Series, Vol. F 68
Neurocomputing
Edited by F. Fogelman Soulié and J. Hérault
© Springer-Verlag Berlin Heidelberg 1990

1. THE ASYNCHRONOUS AND QUADRATIC VERSION (Hinton-Sejnovski-Ackley) :

Let S be the (finite) set of formal neurons or units in such a machine. Each neuron s in S has only two possible states $x_s = 0$ and $x_s = 1$. The interaction between neurons s and t is governed by a synaptic weight w_{st} , and the global configuration $x = (x_s)_{s \in S}$ has an energy

$$G(x) = -\sum_{s,t} w_{st}\, x_s\, x_t \,. \tag{1.1}$$

The weights are assumed to be *symmetric* : $w_{st} = w_{ts}$.

The dynamics of the machine is stochastic, and controlled by a positive parameter T called the temperature. We first describe *the standard asynchronous version of the machine*, as originally introduced by [H.S.A.]. At each instant $n \in \mathbf{N}$, only one of the neurons may attempt to modify its state. Call s_n its index, which is generally preassigned by a deterministic sequence $(s_1 \ldots s_n \ldots)$ visiting periodically all neurons s in S , but which can also be drawn at random in the set S . In either case, whenever the current configuration is x , and the neuron s attempts to modify its state x_s , the new value $\hat{x}_s$ of x_s is selected at random with the probability

$$\left\{\begin{array}{l} P\left(\hat{x}_s = 1 \mid \text{current state } x\right) = \dfrac{e^{\frac{1}{T} U_s(x)}}{1 + e^{\frac{1}{T} U_s(x)}} \\[3ex] P\left(\hat{x}_s = 0 \mid \text{current state } x\right) = \dfrac{1}{1 + e^{\frac{1}{T} U_s(x)}} \end{array}\right. \tag{1.2}$$

where $U_s(x) = \sum_t w_{st}\, x_t$ is the *action potential* of x at site s .

This stochastic dynamics reaches (in the long run) a probabilistic equilibrium which gives to each configuration x the Gibbs probability

$$P(x) = \frac{1}{Z_T} \exp - \frac{G(x)}{T} \,, \tag{1.3}$$

where

$$Z_T = \sum_{y \in \Omega} \exp - \frac{G(y)}{T} \tag{1.4}$$

and Ω is the space of all configurations on S .

The use of the Boltzmann machine as a pattern classifier involves selecting two disjoint subsets D and R of S (*data units* and *response units*) which together constitute the set V of *visible units* in S , while the other units constitute the set of *hidden units* $H = S - (D \cup R)$

The environment provides on the data set D a family of "stimuli" which are random

configurations $\alpha \in A =$
preassigned response con
classification of pattern
the learning process, at le

The goal of learnin
weights $(w_{st})_{s,t \in S}$ in th
presentation. In the "clam
When α is presented,
respectively, while all the
"unclamped" sequence,
configuration α , but all
dynamics above.

For each pair of
simultaneous activity are
direct observation of x_s
by the rule
(1.5)

where ε is "small".

This interesting al
only local computations.
not obvious from a math
for a Kullback distance b
proving convergence, the
back on both of these pro

2. THE SYNCHRON

Boltzmann machi
versions. These statemen
case, the idea of *simultan*
"availability " of highly
the speed increase could

In this synchrono
every neuron s selects a
(1.1). All these simultane
configuration $\hat{x} = (\hat{x}_s)_{s \in}$

configurations $\alpha \in A = \{0,1\}^D$. To each configuration $\alpha \in A$, we want to associate a preassigned response configuration $\beta = F(\alpha) \in B = \{0,1\}^R$, which will achieve the desired classification of pattern α . The map $F : \alpha \rightarrow F(\alpha)$ is assumed known to the superviser of the learning process, at least on a set $A_{ex} \subset A$ of "examples".

The goal of learning algorithms for the Boltzmann machine is to adjust sequentially the weights $(w_{st})_{s,t \in S}$ in the course of alternate periods of "clamped" and unclamped" example presentation. In the "clamped" sequence, one or several examples are presented successively. When α is presented, all the visible units D and R are clamped on α and $F(\alpha)$ respectively, while all the hidden units follow the Glauber dynamics described above. In the "unclamped" sequence, when example α is presented, the data units are tied to the input configuration α, but all hidden and response units evolve freely, according to the stochastic dynamics above.

For each pair of neurons (s,t), the empirical frequencies p_{st} and $\hat{p}_{st}$ of their simultaneous activity are respectively computed during the clamped and unclamped phases by direct observation of x_s and x_t . The current value of the weight w_{st} is then incremented by the rule

$$\Delta w_{st} = \frac{\varepsilon}{T} \left(\hat{p}_{st} - p_{st}\right) \tag{1.5}$$

where ε is "small".

This interesting algorithm derived by [H.S.A.] has the obvious advantage of using only local computations. However, it is highly stochastic in nature, so that *its convergence is not obvious from a mathematical point of view.* Roughly, it approximates gradient descent for a Kullback distance between marginal distribution on the set of visible units. Even after proving convergence, the optimal qualities of the limit are not obvious either. We will come back on both of these problems below.

2. THE SYNCHRONOUS AND QUADRATIC VERSIONS

Boltzmann machines are reputed to be fairly slow as compared to deterministic versions. These statements deserve to be qualified, but we shall not do it in this paper. In any case, the idea of *simultaneously* updating all neurons $s \in S$ is fairly tempting in view of the "availability " of highly parallel computing machines such as the connection machine, since the speed increase could presumably be of the order of Cardinal(S) .

In this synchronous dynamics, at time n , if the current configuration $X(n) = x$, every neuron s selects at random its next state $\hat{x}_s$, according to the probability distribution (1.1). All these simultaneous choices are *independent* of each other. The new state is then the configuration $\hat{x} = (\hat{x}_s)_{s \in S}$.

Contrary to naive expectations, *the long run limit of this stochastic synchronous dynamics* ***is not*** *the Gibbs measure* (1.2) *associated to* G(x) . In fact, *if we define the synchronous energy by*

$$K_T(x) = -\sum_s T\log\left[1+e^{\frac{1}{T}U_s(x)}\right], \tag{2.1}$$

where $U_S(x) = \sum_t w_{st}\, x_t$ is the action potential at site s , then we can state the following proposition.

2.2. PROPOSITION.- *Assume the synaptic weights* w_{st} *to be symmetric. The limit probability distribution for the synchronous dynamics is given by the new Gibbs measure*

$$Q(x) = \frac{1}{\Gamma_T}\exp\left[-\frac{1}{T}K_T(x)\right] \tag{2.3}$$

where

$$\Gamma_T = \sum_{y\in\Omega} e^{-K_T(y)/T}$$

and K_T *is given by* (2.1). *Note that* $K_T(x)$ *is temperature dependent in this formulation.*

The details of the computation will be given elsewhere ; they are similar to previous computations of that type (*cf.* Little [L], Peretto [P], Trouvé [T]). Let us point out a few consequences of formula (2.3).

Call N_S *the set of asynchronous neighbours* of neuron s , namely the set of $t \in S$ such that the local conditional laws verify

$$P(x_s = \lambda \mid x_t, t\in S - s) \equiv P(x_s = \lambda \mid x_t, t \in N_s) \tag{2.4}$$

for $\lambda = 1,0$. Intuitively, the neighbours of s are those which "directly" interact with neuron s .

Then for the asynchronous standard limit distribution P , associated to the energy G , one has obviously

$$N_s = \{t\in S \mid w_{st} \neq 0\}. \tag{2.5}$$

However, for the synchronous limit distribution Q an easy computation shows that the set of synchronous neighbours $\tilde{N}_S$ of neuron s is the union of all the $(N_u - s)$ such that N_u contains s . More precisely

$$\tilde{N}_s = \{t\in S \text{ such that there is a } u\in S \text{ for which } w_{us}\neq 0 \text{ and } w_{tu}\neq 0\}.$$

Thus the cardinal of $\tilde{N}_S$ is generally *larger* than N_S in standard setups.

An easy computation shows that *at very low temperature* T , the synchronous energy is equivalent to

(2.6) $\begin{cases} K_T(x) \sim - \\ K_T(x) \sim (\end{cases}$

On the other hand, *at ver*

(2.7)

One interesting co
equilibrium distribution
neuron t *firing at a cons*
very high temperature, t
achieved.

Also formulas (
asynchronous limit may
This raises the pro
Boltzmann machines. Ac

3. A SUITABLE LE
MACHINE

Assume that the e
distribution ν , so that e
Call $\Omega_V = A \times B$
the machine is coupled t
$Y \in A$ is presented on
equilibrium is reached. C
Ideally, we would
possible to θ where

(3.1) $\begin{cases} \theta(\alpha,\beta) = \\ \theta(\alpha,\beta) = \end{cases}$

Recall that F is the des
has been *replaced* by a pr

(3.2) $\begin{cases} \tilde{\theta}(\alpha,\beta) \\ \tilde{\theta}(\alpha,\beta) \end{cases}$

and $\varepsilon > 0$ is small. The a

$$(2.6)\quad \begin{cases} K_T(x) \sim - \sum\limits_{\{s | U_s(x)>0\}} U_s(x) & \text{if } \{s \mid U_s(x) > 0\} \neq \emptyset \\[2ex] K_T(x) \sim 0 & \text{if } \{s \mid U_s(x) > 0\} = \emptyset \,. \end{cases}$$

On the other hand, *at very high temperature* T, the synchronous energy can be replaced by

$$(2.7)\qquad \tilde{K}_T(x) \sim -\frac{1}{2}\sum_s U_s(x)\,.$$

One interesting consequence of (2.7) is that *at very high temperature, the synchronous equilibrium distribution forces the effective stochastic independence of all neurons, with neuron* t *firing at a constant frequency* $p_t = (e^u)/(1+e^u)$ and $u_t = 1/2 \sum\limits_s w_{st}$. Hence, *at very high temperature, the synchronous machine is in total disorder and no learning can be achieved.*

Also formulas (2.6) (2.7) show clearly that in general the synchronous and asynchronous limit may be extremely different.

This raises the problem of the *feasibility of learning algorithms on the synchronous Boltzmann machines*. Actually, we shall derive a new learning algorithm in this case.

3. A SUITABLE LEARNING ALGORITHM FOR THE SYNCHRONOUS MACHINE

Assume that the environment induces on the set $A = \{0,1\}^D$ *an a priori probability distribution* ν, so that each stimulus $\alpha \in A$ appears with frequency $\nu(\alpha)$.

Call $\Omega_V = A \times B$ the set of all configurations on the visible units $\{D \cup R\}$. When the machine is coupled to the environment through the data units D, a random configuration $Y \in A$ is presented on D, and the machine provides a random response $Z \in B$ when equilibrium is reached. Call $\hat{\theta}$ the probability distribution of the pair (Y,Z) at equilibrium.

Ideally, we would like to select the weights (w_{st}) such that $\hat{\theta}$ becomes as close as possible to θ where

$$(3.1)\quad \begin{cases} \theta(\alpha,\beta) = \upsilon(\alpha) & \text{if } \beta = F(\alpha) \;,\; \alpha \in A \\ \theta(\alpha,\beta) = 0 & \text{if } \beta \neq F(\alpha) \;,\; \alpha \in A \,, \beta \in B \,. \end{cases}$$

Recall that F is the desired classification map. More practically, we shall assume that θ has been *replaced* by a probability $\tilde{\theta}$ on Ω_V such that

$$(3.2)\quad \begin{cases} \tilde{\theta}(\alpha,\beta) = (1-\varepsilon)\,\theta(\alpha,\beta) & \text{when } \theta(\alpha,\beta) \neq 0 \\ \tilde{\theta}(\alpha,\beta) = \varepsilon/M & \text{when } \theta(\alpha,\beta) = 0 \end{cases}$$

and $\varepsilon > 0$ is small. The actual value of ε is irrelevant, and $M = \text{card}\{\Omega - \text{support}\,\theta\}$.

One way of evaluating the distance between $\tilde{\theta}$ and $\hat{\theta}$ is thus to compute the Kullback distance

(3.3) $$d(\tilde{\theta},\hat{\theta}) = -\sum_{\alpha\beta\in A\times B} \tilde{\theta}(\alpha\beta)\log\frac{\hat{\theta}(\alpha\beta)}{\tilde{\theta}(\alpha\beta)} .$$

The vector of optimal weights $w = (w_{st})_{s,t\in S\times S}$ should then minimize $d(\tilde{\theta},\hat{\theta})$. In particular, we should try to have the zero gradient condition

(3.4) $$\text{grad}_w\, d(\tilde{\theta},\hat{\theta}) = 0$$

and a natural gradient algorithm is of course to update current values of w by Δw where

(3.5) $$\Delta w \text{ is a (small) multiple of } \left[-\,\text{grad}_w\, d(\tilde{\theta},\hat{\theta})\right] .$$

This approach can be carried out in the synchronous case, in view of the explicit formula given above for the synchronous energy, and the computations, which will be given elsewhere, present no serious difficulty ; as a particular case of the general formula (6.6) below, one obtains for the synchronous update Δw_{st}

(3.6) $$\Delta_{synch}\, w_{st} = \frac{\eta}{T}\left[E^{clamped}(\gamma_{st}) - E^{unclamped}(\gamma_{st})\right]$$

where η is a small scalar and

(3.7) $$\gamma_{st} = x_t\,\frac{e^{U_s(x)/T}}{1+e^{U_s(x)/T}} + x_s\,\frac{e^{U_t(x)/T}}{1+e^{U_t(x)/T}} .$$

Here $E^{clamped}(f)$ represents the theoretical expected value of the random variable $f(x)$ when all visible units are clamped on the pairs (stimulus, desired response). Similarly, $E^{unclamped}(f)$ represents the same quantity when only the input units D are coupled to the environment.

Note that from a practical point of view, these mathematical expectations $E(f)$ can and must be estimated by natural time averages of $f(x)$ at equilibrium.

To understand better the physical nature of γ_{st} , note that

$$\rho_s(x) = \frac{e^{U_s(x)}}{1+e^{U_s(x)/T}}$$

represents the local probability of replacing x_s by 1 , when the global actual configuration is x , in the natural synchronous updating of all the $(x_u)_{u\in S}$. Hence, these quantities are locally computed by the machine in a systematic fashion. However, using that remark, we can now give an extremely intuitive interpretation of $E(\gamma_{st})$. Call n an arbitrary instant in the time clock of the synchronous machine. Call $X(n)$ the configuration of the machine at time n , and $X_s(n)$ the state of neuron s at time n . Then in the clamped case, as well as in the unclamped case, we have

(3.8) E

and thus the interesting fo

(3.9)

where q_{st} *is the probab*

meaningful local quant

(equilibrium) probabiliti

shall keep *the notation* q

case.

The natural time

frequencies of delayed co

(3.10) $\frac{1}{1+N}\left[X_s(n)\right.$

which point out *the rele*

synchronous activity of th

We thus propose *th*

(3.11)

where $q_{st}\, q_{ts}\, \hat{q}_{st}\, \hat{q}_{ts}$ are

Note that the computation

The mathematical

recommends the use of *c*

make convergence possib

(3.12)

where c,b remain const

current weights updating.

4. CONVERGENCE

It turns out that the

asynchronous is quite sin

observed Markov fields.

Younes in his doctorate t

Here of course the

and hence may be high-

function $\phi(w) = d(\tilde{\theta},\hat{\theta})$, (3.8) $$E(\gamma_{st}) = \lim_{n\to\infty} E\left[X_s(n)\,X_t(n+1) + X_t(n)\,X_s(n+1)\right]$$

and thus the interesting formula

(3.9) $$E(\gamma_{st}) = q_{st} + q_{ts} ,$$

where q_{st} *is the probability of successive firing for neuron* s *and neuron* t . Hence, the meaningful local quantities in synchronous Boltzmann machines turn out to be the (equilibrium) probabilities q_{st} of *delayed co-firing* for ordered pairs (s,t) of neurons. We shall keep *the notation* q_{st} *for the clamped case* and use *the notation* $\hat{q}_{st}$ *for the unclamped case.*

The natural time average estimates for q_{st} , $\hat{q}_{st}$ are of course the empirical frequencies of delayed cofiring (at equilibrium)

(3.10) $$\frac{1}{1+N}\left[X_s(n)\,X_t(n+1) + X_s(n+1)\,X_t(n+2) + \ldots + X_s(n+N)\,X_t(n+N+1)\right]$$

which point out *the relevance of delayed correlations between neighboring neurons in the synchronous activity of the machine.*

We thus propose *the following natural synchronous learning algorithm*

(3.11) $$\Delta_{\text{synch}}\, w_{st} = \frac{\eta}{T}\left[(q_{st} + q_{ts}) - (\hat{q}_{st} + \hat{q}_{ts})\right]$$

where $q_{st}\, q_{ts}\, \hat{q}_{st}\, \hat{q}_{ts}$ are estimated by the empirical frequencies of delayed cofiring (3.10). Note that the computations involved are purely local and extremely simple.

The mathematical results on stochastic gradient algorithms ([B.M.P.] [Y]) strongly recommends the use of *a slowly decreasing coefficient* η (the "gain" of the algorithm) to make convergence possible. A mathematically classical choice is

(3.12) $$\eta(k) = \frac{c}{b+k}$$

where c,b remain constant during learning and the integer $k \in \mathbf{N}$ denotes the *index of the current weights updating.*

4. CONVERGENCE OF THE LEARNING ALGORITHMS

It turns out that the learning problem for Boltzmann machines wether synchronous or asynchronous is quite similar to the problem of maximum likelihood estimation for partially observed Markov fields. Thus most of the sophisticated probabilistic techniques used by Younes in his doctorate thesis (*cf.* [Y]) are relevant in this context.

Here of course the parameter to be discovered is the unknown vector of weights w and hence may be high-dimensional. Consider again the problem of minimizing in w the function $\phi(w) = d(\tilde{\theta},\hat{\theta})$, defined by (3.3).

As is easily checked, the function $w \to \phi(w)$ is often *not convex*, and hence the stochastic gradient algorithms proposed for learning in synchronous or asynchronous Boltzmann machines may very well be trapped in local minima of ϕ. Note here that our statements concern learning at fixed temperature which we consider as quite relevant in artificial vision applications for instance, as will be explained elsewhere.

In generic situations, the theory of stochastic gradient algorithms following the lines of [B.M.P.] and [Y], can thus only prove quasi-convergence of learning algorithms. This means essentially that if the successive learned values $w(k)$, $k = 1,2,\ldots$ of the vector of weights come back infinitely often within any single well of the energy landscape associated to the function ϕ, then the sequence $w(k)$ becomes ultimately trapped at the bottom of that well. Hence for all practical purposes, either the sequence $w(k)$ explodes, or it is bound to achieve *local* optimisation, in generic Boltzmann machines.

Another result of the same kind is that if the starting point $w(0)$ for the vector of weights lies close enough to a non degenerate absolute minimum $\tilde{w}$ of $\phi(w)$, and if the coefficient η regulating the gain is small enough, then the learning algorithm will converge towards the absolute minimizing weight vector $\tilde{w}$. From a practical point of view, this last point means that *small variations of the environment characteristics can be successfully corrected by learning once the machine has initially been properly tuned.*

As for learning using decreasing temperature schedules, it can be studied through similar stochastic techniques, but we shall come back to this question elsewhere.

5. GENERAL BOLTZMANN MACHINES WITH MULTIPLE INTERACTIONS

In the standard quadratic Boltzmann machines, interaction between neurons is limited to *pairs of neurons*. Actually, one can usefully develop a learning theory for much more geenral stochastic machines, within the formalism of Gibbs measures.

As a first and innocuous generalizing step assume that each neuron may now take a finite family of values $\lambda \in \Lambda$ instead of the only values $0,1$. Consider an *arbitrary family* Γ of subsets C of S. Each C in Γ will be called a *clique of neurons*, and its degree of activity will be quantitatively measured by an *interaction potential* $J_C(x) = J_C(x_s \ldots x_s)$ if $C = \{s_1 \ldots s_k\}$. These interaction potentials are *arbitrary numerical functions of the clique configuration* x_C.

For each clique $C \subset \Gamma$ introduce a numerical parameter $w_C \in \mathbf{R}$ which will be called the *clique weight*. The vector $w = \{w_C\}_{C \in \Gamma}$ will be called the *weights vector* and will define completely the generalized Boltzmann machine.

Indeed we introduce the asynchronous energy function

$$G(x) = \sum_{C \in \Gamma} w_C J_C(x) \tag{5.1}$$

which measures the *weighted activity of the cliques* to compute the global activity of the configuration. A classical Glauber type of asynchronous stochastic dynamics can now be defined to ensure that the asynchronous machine has at equilibrium (in the long run) a Gibbs probability distribution on the set Ω of configurations given by

$$P(x) = \frac{1}{Z_T} e^{-G(x)/T} , \tag{5.2}$$

where Z_T is as before the partition function, and T a positive fixed temperature parameter.

Define the set N_S of neighbours of neuron s as the set of all $t \in S$ such that $t \neq s$ and there is some clique $C \in \Gamma$ containing both s and t. Then the *local action potential* at s is defined as $U_S(x) = U_S(x_S, x_N)$ by

$$U_s(x_s, x_{N_s}) = - \sum_{\{C \ni s\}} w_C J_C(x) . \tag{5.3}$$

And the proper random updating of a single neuron s is here to select the next state $\hat{x}_S \in \Lambda$ with the probability

$$p(\hat{x}_s) = \frac{e^{U_s(\hat{x}_s, x_{N_s})/T}}{\sum_{\lambda \in \Lambda} e^{U_s(\lambda, x_{N_s})/T}} . \tag{5.4}$$

To compare this setup with the standard asynchronous quadratic Boltzmann machine, take the set Γ of active cliques to coincide with the set of arbitrary *pairs* of neurons, and interaction potentials

$$J_C(x) = J_{\{s,t\}}(x) = -2 \, x_s \, x_t . \tag{5.5}$$

Then the weight w_C of the clique $C = \{s,t\}$ is simply the synaptic weight w_{st}, and the random updating (5.4) is exactly the one we recalled earlier in (1.2).

The problem is now to define a suitable learning algorithm for the asynchronous generalized Boltzmann machine. To do this, we have to compute the distance $\phi(w) = d(\tilde{\theta}, \hat{\theta})$ between clamped and unclamped distributions on Ω_V and evaluate the gradient $\partial\phi/\partial w$.

A direct computation shows that one has

$$T \frac{\partial \phi}{\partial w_C}(w) = E_{\text{clamped}} \left[J_C \right] - E_{\text{unclamped}} \left[J_C \right] \tag{5.6}$$

and thus the *natural learning algorithm for asynchronous general Boltzmann machines* consists in updating the vector w of clique weights by

$$\Delta w_C = \frac{\eta}{T}\left\{E_{\text{unclamped}}(J_C) - E_{\text{clamped}}(J_C)\right\} \tag{5.7}$$

where the (small) gain η decreases slowly at the rate $\eta(k) = c/(b+k)$ where k is the weight update index.

Of course in the now classical quadratic case, for clique $C = \{s,t\}$, the expected value of $1/2\ J_C = -\ x_s\, x_t$ is simply $(-\ p_{st})$ where p_{st} is the probability of simultaneous firing for neurons s and t , at equilibrium. Practically a good estimate of $E(J_C)$ is provided by *a direct time average of the clique activity* $J_C(x)$. Thus, *computations remain local* and just as feasible as in quadratic machines.

The network should then be viewed as a double network $S \cup \Gamma$ where to each index $C \in \Gamma$ is associated a cell or processor which we call a *clique indicator* and whose states belong to a finite subset of $\mathbf{R}$. Namely, whenever the network S is in configuration x , *the state* y_C *of the clique indicator* C is

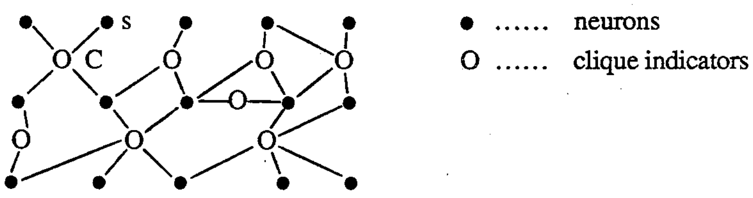
$$y_C = J_C(x)\ . \tag{5.8}$$

The connection between neuron $s \in S$ *and clique indicator* $C \in \Gamma$ *exists if and only if* s *belongs to the clique* C . Moreover, this connection is simply a *message transmission channel.* From s to C , it transmits the state x_s to the processor C , which once it has gotten hold of all the $(x_s)_{s\in C}$ computes deterministically its state $y_C = J_C\ (x_s\, ,\, s \in C)$.

Conversely from C to s , the connection transmits the state y_C , that is the corresponding cell activity, with the weight w_C . Then by a simple sum over all cliques C connected to s , the neuron s can compute its action potential

$$U_s(x) = -\sum_{C \text{ containing } s} w_C\, y_C \tag{5.9}$$

and then neuron s uses this number for its own random updating. Several detailed setups corresponding to this random update can easily be proposed and will be described elsewhere.



Example of a small network with cliques of various cardinals

The configuration x of S thus provides a direct computation (a deterministic one) of the configuration y of Γ , which in turn permits a proper mechanism for random updating of x.

The updating of t
indicator activity, a simp
C.

Hence the asynch
learning tool, in pattern r
have been studied throug
D. and S. Geman (see [A
suitably reinterpreted i
generalized Boltzmann
detection and segmentati

6. THE SYNCHRON

For a general Bolt
much more difficult than
not linked to the fact t
interaction of order ≥
equilibrium distribution.
shall limit ourselves to t
interactions only.

Call $J_{st}(x_s,x_t)$ th
weight of the clique $\{s,t$
the synchronous energ
microbalance equations,

(6.1)

with synchronous equilib

(6.2)

One can now check that
delayed interactions

(6.3) $d_{st} + d_{ts}$

where $X(n)$ is the globa

These average del
interactions

The updating of the weights requires essentially a time monitoring of the clique indicator activity, a simple computation which can be done by a local processor included in C.

Hence the asynchronous generalized Boltzmann machine looks like an interesting learning tool, in pattern recognition. Indeed in artificial vision, many tasks of low-level vision have been studied through Markov field modelizations following the breakthrough papers of D. and S. Geman (see [A] [C] [G] [GG] among many others). Many of these models can be suitably reinterpreted in the context we have just described, to design asynchronous generalized Boltzmann machines which achieve low-level vision tasks such as contour detection and segmentation. We shall come back to this exciting topic in another paper.

6. THE SYNCHRONOUS GENERAL BOLTZMANN MACHINE

For a general Boltzmann machine, the study of the synchronous equilibrium energy is much more difficult than in the case of standard Boltzmann machines. The difficulty here is not linked to the fact that we abandon quadratic forms, *but to the existence of multiple interaction of order* ≥ 3 , which prevents the explicit computation of the synchronous equilibrium distribution. We have recently solved this problem in [A]. In this brief note, *we shall limit ourselves to the much easier case of general Boltzmann machines with pairwise interactions only.*

Call $J_{st}(x_s,x_t)$ the interaction potentials corresponding to pairs $\{s,t\}$ and w_{st} the weight of the clique $\{s,t\}$. The action potential $U_s(x_s,x_{S-s})$ is given by (5.3) as before and the synchronous energy $K_T(x)$ for this Boltzmann machine can be computed using microbalance equations, to show that

$$K_T(x) = -T\sum_{s}\log\left[\sum_{\lambda\in\Lambda} e^{U_s(\lambda,x_{S-s})/T}\right] \tag{6.1}$$

with synchronous equilibrium distribution

$$Q(x) = \frac{1}{\tilde{Z}_T}\, e^{-K_T(x)/T} . \tag{6.2}$$

One can now check that the gradient of the Kullback distance $d(\widetilde{\theta},\hat{\theta})$ involves *the average delayed interactions*

$$d_{st} + d_{ts} = \lim_{n\to\infty} E\left\{J_{st}\left[X_s(x),X_t(n+1)\right] + J_{st}\left[X_s(n+1),X_t(n)\right]\right\} \tag{6.3}$$

where $X(n)$ is the global neuron configuration at time n .

These average delayed interactions can be correctly estimated by *empirical delayed interactions*

$$\frac{1}{N}\left\{J_{st}\left[X_s(n),X_t(n+1)\right]+\ldots+J_{st}\left[X_s(n+N-1),X_t(n+N)\right]\right\}, \tag{6.4}$$

which we still denote by d_{st} in the clamped case and $\hat{d}_{st}$ in the unclamped case.

Then one can prove that the synchronous learning algorithm by gradient descent must compute the update Δw_{st} of the clique weight w_{st} by

$$\Delta w_{st} = \frac{\eta}{T}\left[(\hat{d}_{st}+\hat{d}_{ts})-(d_{st}-d_{ts})\right] \tag{6.5}$$

with a slowly decreasing gain η, as before in (3.12). Recall that in the standard quadratic case $1/2\ J_{st}(x_s\ x_t)$ is $(-\ x_s\ x_t)$ so that these formulas do generalize those of § 3.

7. FURTHER EXTENSIONS

One may want to evaluate the clique activity from several quantitative points of view and weight these points of view according to their relative importance. This is handled by vector valued interaction potentials J_C and clique weights w_C.

It is also possible to imagine more realistic partially synchronous updates, in which at each tick of the clock, only a fixed proportion of the neurons S are drawn at random and allowed to perform simultaneous random updates. As long as the probability of having 3 or more simultaneous updates within a single active clique is kept small, the theory of learning for partially synchronous updates should strongly resemble the totally synchronous update for pairwise interacting machines. Hence, in such cases, learning by using average delayed interactions as described above should still perform a good gradient descent.

BIBLIOGRAPHY

[A] R. AZENCOTT - *Markov fields and low-level vision tasks*. Proc. Int. Cong. App. Math., ICIAM, Paris (1987).
- *Gibbs fields, simulated annealing, and low-level vision tasks*, Proc. Congress on Pattern Recognition AFCET - INRIA, Antibes (1987).
- *General Boltzmann machines with multiple interactions* (to appear in IEEE PAMI, 1990).
- *Parameter estimation for synchronous Markov fields* (to appear 1990).

[B.M.P.] A. BENVENISTE, M. MÉTIVIER, R. PRIOURET - *Algorithmes stochastiques*, Masson, Paris (1988).

[C] B. CHALMOND - *Image restoration using an estimated Markov model*, IEEE PAMI (1988).

[G] D. and S. GEMAN - *Gibbs fields, simulated annealing, and Bayerian reconstruction of images*, IEEE PAMI (1984).

[G.G.] S. GEMAN an
Int. Cong. Mat

[H.S.A.] G. HINTON, T
constraint satis
Univ. (1984).

[L] W.A. LITTLE

[P] P. PERETTO -

[T] A. TROUVÉ -
preprint (1988

[Y] L. YOUNES -
(1988) and doc

$),X_t(n + N)]\}$,

nclamped case.

by gradient descent must

.t in the standard quadratic

:e those of § 3.

quantitative points of view

ırtance. This is handled by

:onous updates, in which at

are drawn at random and

; probability of having 3 or

mall, the theory of learning

totally synchronous update

ʒ by using average delayed

nt descent.

n tasks. Proc. Int. Cong.

, and low-level vision tasks,
ꝚIA, Antibes (1987).

multiple interactions (to

ıous Markov fields (to

Algorithmes stochas-

ted Markov model, IEEE

g, and Bayerian reconstr-

[G.G.] S. GEMAN and C. GRAFFIGNE - *Gibbs fields and image segmentation,* Proc. Int. Cong. Math. (1987).

[H.S.A.] G. HINTON, T. SEJNOWSKY, D.H. ACKLEY - *Boltzmann machines : constraint satisfaction networks that learn,* Technical report Carnegie Mellon Univ. (1984).

[L] W.A. LITTLE - *The existence of persistent states,* Math. Biosci 19 (1974).

[P] P. PERETTO - *Collective properties of neural networks*, Preprint (1984).

[T] A. TROUVÉ - *Parallelization of simulated annealing,* C.R. Ac. Sci. (1988) and preprint (1988).

[Y] L. YOUNES - *Parameter estimation for Gibbs fields*, Ann. Inst. H. Poincaré (1988) and doctorate thesis Univ. Paris-Sud (1988).