THERMODYNAMIC FORMALISM FOR THE STUDY AND FORMATION OF ALGORITHMS AND NEURAL NETWORKS

I.F. YASINSKIY[1] & F.N. YASINSKIY[2]
[1]Ivanovo State Textile Academy, Russia.
[2]Ivanovo State Power University, Russia.

ABSTRACT

This article considers the questions connected with creation of optimum algorithms using the laws of thermodynamics as applied to a computing process. Ideas and methods of phenomenological and statistic thermodynamics are used to estimate the amount of calculations or volume of the neural network. Introduction of the other thermodynamic functions, besides entropy, and also definition of the three thermodynamic origins in the context of calculations allow to study stability, organize the parameters according their information weights, carry out the decomposition of the complex systems, construct the rapid algorithms. The way of creation of the neural network structure is offered, consisting in use of the pre-trained fragments.

Keywords: Computational entropy, neural network structure, rapid algorithms, thermodynamics of calculations.

1 INTRODUCTION

Construction of optimal algorithms, that can minimize the cost of computer time with the definite accuracy of results, is an important and actual task. Earlier Hinchin [1] and Kolmogorov [2] introduced the concept of computational entropy. It allows to get closer to estimation of the computations complexity in some problems. After that a variety of researches appeared [3–6], where the entropy was effectively used for the complexity evaluation of algorithms and programs.

In this work the other thermodynamic functions, besides entropy, are introduced: pressure as costs in the number of operations for the dimension change, temperature as cost related to the extracted information, potential as change of operations number in connection with obtaining the answer with a certain probability. Some applications of the offered ideas are described.

2 THERMODYNAMIC FORMALISM

It is offered to study algorithms and neural networks structures by means of the following thermodynamic formalism.

Suppose, there is an iterative computational process:

$$x^{k+1} = F\left(x^k, \xi^k\right), \tag{1}$$

where $x^k \equiv \left(x_1^k, x_2^k, ..., x_n^k\right)$ – point in the n-dimensional phase space X, F – some algorithm, k – number of iteration, ξ^k – noise considering parameter.
If the process converges, then

$$x_i^k \to x_i^*, k \to \infty, i = 1, 2, ..., n, \tag{2}$$

where x^*– final point, $x^* \in X$.

ISSN: 2046-0546 (paper format), ISSN: 2046-0554 (online), http://journals.witpress.com
DOI: 10.2495/CMEM-V1-N4-456-465

The array of phase points with density $\rho(x_1,...,x_n)$ is taken in X. This array may be considered as a sort of 'phase gas', which compresses to the point x^* during calculations.

It is offered to observe this process from the thermodynamic point of view. The first results in this direction belong to A. N. Kolmogorov and his research group. They introduced the concept of computational ε – entropy and studied its properties [1,7]. Also, interesting researches in this direction were carried out by Babenko [8], Aho et al. [9], Traub and Voznyakovskiy [10], Fuller and Reyzin [11]. Below the other thermodynamic functions are introduced and thermodynamic postulates for the computational process formulated.

The first law of thermodynamics:

$$dN = T \cdot dI + p \cdot dn + \mu \cdot d\pi, \tag{3}$$

where N – number of operations, taken for the 'phase gas' compression, N – analog of the internal energy (has the additivity property), I – information, received in the process of calculations.

Algorithmic temperature:

$$T = \left.\frac{\partial N}{\partial I}\right|_{n,\pi}, \tag{4}$$

is a number of operations, taken for obtaining of the information unit.

Pressure:

$$p = \left.\frac{\partial N}{\partial n}\right|_{I,\pi}, \tag{5}$$

represents the number of operations, taken for dimension number change, if it occurs during calculations.

Summand $\mu \cdot d\pi$ in eqn (3) takes into account the probabilistic character of the calculation process (errors, use of random numbers). Here:

$$\mu = \left.\frac{\partial N}{\partial \pi}\right|_{I,n}, \tag{6}$$

π – probability of a correct solution, $0 < \pi < 1$. In some cases the definite decrease of π leads to a substantial decrease of the operations number N (methods Monte-Carlo, random search).
The second law of thermodynamics. Thermodynamic irreversibility.

The calculation process is stable, if

$$dI > 0. \tag{7}$$

Using this inequality, algorithms can be tested for stability. This is much more efficient unconventional way, it does not require linearization and is applicable in complex nonlinear cases.

Obtained information:

$$\Delta I^{k,k+m} = H^k - H^{k+m}, \tag{8}$$

where H^k, H^{k+m} – calculation entropy on the steps k and $k+m$.
The third law of thermodynamics. Unattainable zero temperature:

$$T = \left.\frac{\partial N}{\partial I}\right|_{n,\pi} > 0. \tag{9}$$

The value, inverted to the algorithmic temperature, is also very interesting. It is called the output of the algorithm and is written as the obtained information divided by the number of operations:

$$C^{K_1,K_2} = \frac{\Delta I^{K_1,K_2}}{\Delta N^{K_1,K_2}}. \tag{10}$$

Variational problem of search of the optimal distribution of the 'phase gas' initial density, that minimize the number of operations ΔN for specific information ΔI leads to the L.Boltzmann's distribution:

$$\rho^0(x_1,...,x_N) = \exp\left(-\frac{\Delta N}{T}\right)\cdot\left(\sum_{i=1}^{n}\exp\left(\frac{\Delta N}{T}\right)\right)^{-1}. \tag{11}$$

Hence it is elementary to obtain all the known relations of statistical thermodynamics, concordant with the described phenomenological approach. If the local linearization of the mapping eqn (1) is made according to the method of Newton:

$$y_i^{k+1} = \sum_{j=1}^{n} A_{ij} y_i^k + \zeta_i^k, \tag{12}$$

$$y_i = x_i - x_i^*,\ A_{ij} = \frac{\partial F_i}{\partial x_j},$$

and the information of Hartly is inserted:

$$\Delta I^{o,k} = -\ln\frac{V^k}{V^o}, \tag{13}$$

where V^o, V^k – initial and final volumes in the phase space X, then can be shown, that:

$$\Delta I^{o,k} = -\sum_{i=1}^{n}\ln\left(\lambda_i^k + \chi_i\frac{1-\lambda_i^{k-1}}{1-\lambda_i}\right), \tag{14}$$

where λ_i – eigenvalues of matrix A, χ_i – constants, proportional to the noise level for the frequencies λ_i. In highly conditional case, if taken $\lambda_s = \lambda$; $\chi_s = \chi$; $s = 1,2,...,n$, and consider the calculation costs proportional to $k \cdot n(\nu + n)$, then the expression can be written:

$$p(\nu + n) = I \cdot T, \tag{15}$$

which can be called as the equation of the algorithm state. In more complex cases the equations of state will not be so simple.

The conception of thermodynamic cycles is applicable to iterative calculation processes. As example the minimum search by using the gradient method can be considered:

$$x_i^{k+1} = x_i^k - \tau\left.\frac{\partial Q}{\partial x_i}\right|_{x^k}, \tag{16}$$

Table 1: Carnot cycle for the optimization process.

Sector	Temperature initial	Temperature final	Information initial	Information final
1	T_2	T_1	I_1	I_1
2	T_1	T_1	I_1	I_2
3	T_1	T_2	I_2	I_2
4	T_2	T_2	I_2	I_1

comparing it to a Carnot cycle in the coordinates (I,T) – Table 1.

Let $0 < I_1 < I_2$, $0 < T_1 < T_2$.

Sector 1. Gradient components are calculated, giving the direction 'on target'. The output of the algorithm increases, temperature decreases.
Sector 2. Move toward the target according the obtained direction. Entropy falls down, information grows.
Sector 3. The calculated gradient components have no value at the new position of the searching point. They are discarded. The output decreases and temperature grows.
Sector 4. The decision is made to use the same method in the new point. Return to the starting point. The cycle (I, T) closed up on the diagram.

Besides stability evaluation the described formalism allows to find the optimal ways of decomposition, allocate the major variables, reduce the task dimension, effectively randomize calculations.

The above formulation can be used in relation to neural networks. In this instance instead of the operations number N the value ε is used, that is proportional to the energy diffusion of the neural network:

$$\varepsilon = b_n \cdot n + b_m \cdot m, \tag{17}$$

where n – neurons quantity, m – number of connections between neurons, b_n, b_m – energies, dissipated respectively by neurons and connections.
Expressions (3), (4), (5), (6) may be rewritten in the form:

$$d\varepsilon = T \cdot dI + p \cdot dn + \mu \cdot d\pi, \tag{18}$$

$$T = \left.\frac{\partial \varepsilon}{\partial I}\right|_{n,\pi}, \; p = \left.\frac{\partial \varepsilon}{\partial n}\right|_{I,\pi}, \; \mu = \left.\frac{\partial \varepsilon}{\partial \pi}\right|_{I,n,m}. \tag{19}$$

If supposed, that all neurons work in the areas of the quickest transitions, where the linearization is locally admissible, then expression (14) can be used:

$$\Delta I^{o,k} = -\sum_{i=1}^{n} \ln\left(\lambda_i^k + \chi_i \frac{1-\lambda_i^{k-1}}{1-\lambda_i}\right). \tag{20}$$

Here k – serial number of the neuron layer during the direct passage (from input to output). Based on this, it is possible to formulate the problems for the selection of principal components, simplification, network decomposition according the weakest informational connections,

transition to the new variables. Presence of the summand $\mu \cdot d\pi$ allows to suppose, that processes randomization in the neural network using Markov algorithms may be effective.

3 APPLICATIONS OF THE THERMODYNAMIC FORMALISM

3.1 Calculation process stability

In this section some applications of the thermodynamic approach are considered. First of all, about the stability of calculation process. Suppose there is a multi-step iterative process:

$$\vec{x}^{k+1} = F\left(\vec{x}^{k}, \vec{x}^{k-1}, ...\right). \tag{21}$$

Here, $\vec{x} \equiv \left(x_1, ..., x_N\right)$ – point in the N-dimensional phase space, k – number of iteration.

As said before, a variety of initial points X, $\vec{x}^0 \in X$, which can be considered as some kind of the phase gas, move according eqn (21). In the phase space it is possible to set the grid structure, consisting of similar N-dimensional cubic cells, and according the number of particles of phase gas, caught in each cell, to evaluate its density ρ^k. It allows to trace the entropy change on each step of the calculation process Vatolin [12]:

$$H^k = -\int_X \rho^k \ln \rho^k \cdot dx_1 .. dx_N . \tag{22}$$

Entropy reduction, and, hence, information extraction, testifies about calculations stability. Such approach to the stability evaluation surpasses the traditional one, based on linearization and eigenvalues analysis. For example, the display

$$U^{k+1} = \left(U^k\right)^{\frac{1}{m+1}}; U^{k+1} \to 1 \text{ at } k \to \infty \tag{23}$$

is, obviously, stable upon condition $0 < m < \infty$. At the same time the traditional approach shows instability in described case. This was shown by Vatolin [13] for $m = 1$.

3.2 Dimension change

Now about the change of dimension. If the technology of the principal component analysis is used Haykin [14] and there is passing from coordinates $\left(x_1, ..., x_N\right)$ to new $\left(z_1, ..., z_N\right)$, then it is possible to define information weight of each new coordinate as

$$g_i^k = \ln\left(\lambda_i^k + \chi_i \frac{1 - \lambda_i^{k-1}}{1 - \lambda_i}\right). \tag{24}$$

Here, λ_i^k , χ_i^k – eigenvalues and noise levels on the interval of k-steps. The coordinates with low weights g_i^k can be rejected, that means the dimension reduction. If it will be possible to achieve good conditionality $\lambda_i = \Lambda$ and low noise level $K_i = 0$, then, under preservation of the information quantity, the lowest dimension number of the system may be reached:

$$m = \sum_{i=1}^{N} \frac{g_i}{\ln \Delta} < N. \tag{25}$$

However, the principal components method is a rather expensive operation. In some cases, when operating only the coordinates x_i Kovalevskiy and Reshetnik [15], the rough estimation of their weight will be recorded as:

$$l_i = \ln\left(\sum_{j=1}^{N}\left|\frac{\partial F_i}{\partial x_j}\right|\right). \tag{26}$$

The approach, based on the weights introduction, allows not only to reduce the dimension, but also to carry out the system decomposition. This applies both to internal, and to external connections. In case of a neural network the connections between neurons are meant. Decomposition to blocks is conducted on the weakest connections with the smallest weights. For each block the temperature is introduced (it is measured by number of operations on unit of taken information). Calculations begin with the block with the lowest temperature and finish when the temperatures of all units are equalized and the required information is taken.

3.3 Improving of the neural network structure

An important way to speed up the calculations is the using of priori and posteriori information. For each task the set of related tasks is constructed, which can be a source of information for the main task. Refraining from strict determination of likeness, it should be noted only, that equations for the related problems can be received by means of strikeouts or additions of some components to the initial equation of the main problem. Solutions of related tasks can be used at creation of constructions, in the form of which the solution of original task is looked for. These constructions include constants, which should be chosen optimally for the best approximation. In case of neural networks it consists in inclusion of the pre-trained fragments in the network.

Standard neural network architecture is shown on the Fig. 1. Usually, it has several layers of cells: R – receptor layer, which receives the input data; A – associative layer, neurons of this layer interpret the information; E – effector layer, giving the reaction, or answer, of neural network.

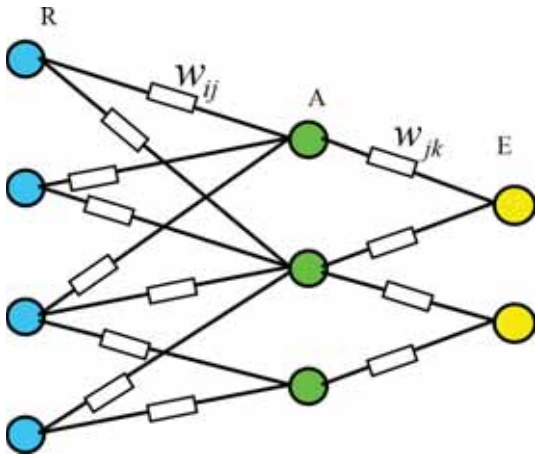

Figure 1: Standard neural network architecture.

The following structure of a neural network should prove to be a productive (Fig. 2). Such neural network at a stage of training contains fragments with the adjusted synoptic connections, previously obtained at the solution of tasks, conjugated with the original. According the offered scheme, the input image or its part is analyzed by the pre-trained blocks. Weights within the blocks do not change during the training process. Only the free weights $w_1, w_2, \ldots, w_N$ on exits of fragments are to be set up. They may be received by means of the optimization algorithms (random search, genetic algorithm, inertial search). The answer on a network exit *Ot* may be given in the form of linear combination of pre-trained fragments answers:

$$Ot = w_1 \cdot Ok_1 + w_2 \cdot Ok_2 + \ldots + w_M \cdot Ok_M = \sum_{i=1}^{M} w_i \cdot Ok_i, \tag{27}$$

where Ok_i – signal on the output neuron of the i-th neural fragment; w_i – weighting coefficient, concerning the influence of the *i*-th fragment on the final answer.

It is also possible to activate the response according:

$$Ot' = \frac{1}{1 + e^{-\frac{Ot}{H}}}, \tag{28}$$

where *H* – constant, that determines the parameters of sigmoid activation function.

Numerical experiments were made on training of a neural network to forecasting of stochastic periodic function *y* eqn (29):

$$y_j = \sum_{i=1}^{N} A_i \cdot \sin(w_i \cdot t_j + \phi_i), \quad i = 1, \ldots, N, \tag{29}$$

$$A_i = \frac{1}{1 + i^2}, \tag{30}$$

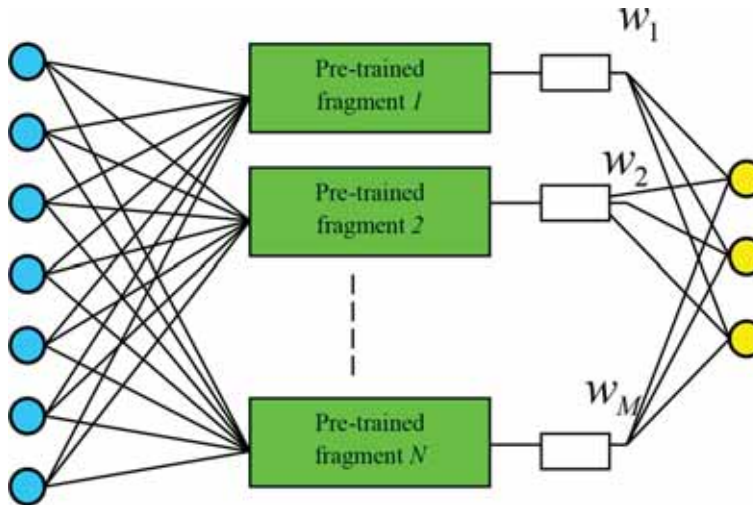

Figure 2: Neural network structure with the pre-trained fragments.

$$w_i = \frac{2 \cdot \pi \cdot i}{N}, \tag{31}$$

$$t_{j+1} = t_j + \tau, \tag{32}$$

where y_j – value of function at the moment of time t_j , A_i – amplitude, w_i – frequency, φ_i – phase (random value), N = 20 – number of summands.

For comparison, the standard network architecture (Fig. 1) and the neural structure that includes the pre-trained fragments (Fig. 2) are taken.

Training pairs were created as follows. Ten consecutive values of function $y_j,\ldots,y_{j+9}$ were set on elements of an input layer of a neural network at parameter t_j change with the fixed step τ. As the true answer on an output layer value of function y_{j+10} acted at the moment t_{j+10}. Thus, the first training pair would be written in the form $\{y_1,\ldots,y_{10}; y_{11}\}$, second – $\{y_2,\ldots,y_{11}; y_{12}\}$ and so on. For training of weights of the network's fragments the similar function with other value of time step τ^* was used. Obviously, by setting various values τ it is possible to receive various sequences y_j, connected, nevertheless, by the conjoint periodic law. Randomnicity of process is characterized by a change range of φ .

Processes of training of a neural network with standard architecture and networks with inclusion of pre-trained fragments are provided on Fig. 3. At comparison of the presented schedules it is visible that training of a neural network with the adjusted fragments occurs quicker, and even at long training the neural network of standard architecture doesn't reach the same level of the right answers.

All appearances, accumulation of a database of the trained neural networks, allowing to solve various problems, will be a further improvement of neural network technologies

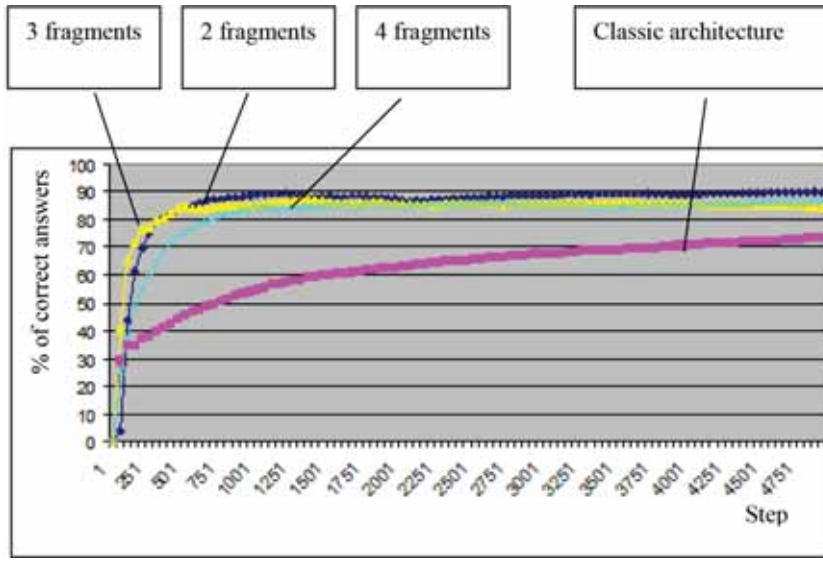

Figure 3: Schedules of training of a neural network with standard architecture and networks with inclusion of the pre-trained fragments.

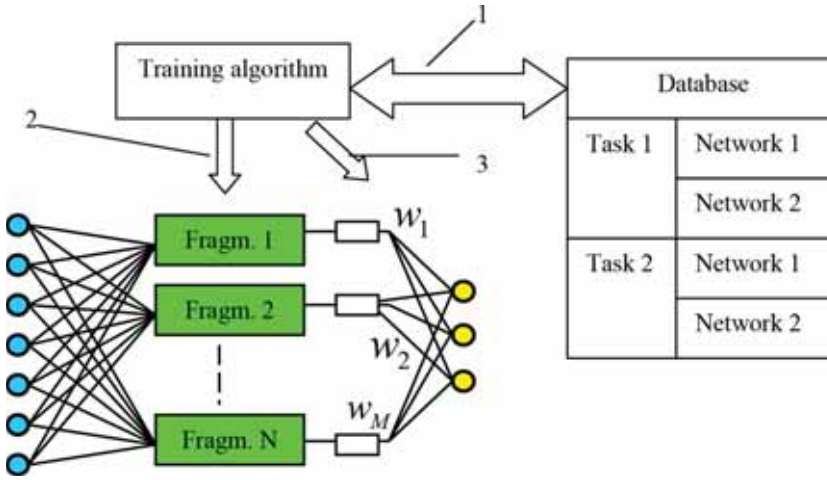

Figure 4: Possibilities of improvement of the neural networks structure.

(Fig. 4). In the process of the new problem solving the training algorithm will choose optimal fragments from database to upbuild the effective neuronetwork structure.

1. Search for optimal solutions to the problem in the fragments database;
2. Structuring of selected fragments;
3. Weight training of each fragment in the resulting system response.

4 CONCLUSION

Thermodynamic formalism opens up new possibilities in building of the optimal algorithms for modeling of the complex dynamic systems and construction of the effective neural networks. The use of thermodynamic laws and functions (temperature, pressure, potential, etc.) introduced for consideration of the computational process stability and system dimensionality reduction. The way of creation of the neural network structure is offered, consisting in use of the pre-trained fragments. Training of such system consists in finding of the connections weights, considering influence of each fragment on the resultant answer. The new method allows to increase the speed of training of a neural network. The described ideas and methods will find certain applications.

REFERENCES

[1] Hinchin, A.Y., *The concept of entropy in probability theory*, UMN, tome 11, **3(55)**, 1953.
[2] Kolmogorov, A.N. & Tihomirov, V.M., *ε -entropy and ε -capacity of sets in functional spaces*, UMN, tome 14, **2(86)**, pp. 3–80, 1959.
[3] Vitushkin, A.G., *Assessment of the Tabulation Problem*, Fizmatiz: Moscow, 1959.
[4] Petri, N.V., *Complexity of the algorithms and time of the their work*, DAN USSR, pp. 30–31, 1969.
[5] Barak, B., Shaltiel, R. & Wigderson, A., Computational analogues of entropy. *Proc. of the 11th International Conference on Random Structures and Algorithms*, pp. 200–215, Springer, 2003.

[6] Haitner, I., Reingold, O., Vadhan, S. & Wee, H., Inaccessible entropy. *Proc. of the 41st Annual ACM Symposium on Theory of Computing (STOC `09),* pp. 611–620, 31 May–2 June, 2009.
[7] Kolmogorov, A.N., The different approaches to the difficulty estimation of the approximate setting and computing of functions. *Proc. of the Int. Congress of Mathematicians*, Stockholm, 1964.
[8] Babenko, K.I., *Theoretical Foundations and Construction of Numerical Algorithms for Problems of Mathematical Physics*, Nauka: Moscow, p. 295, 1979.
[9] Aho, A., Hopkroft, J. & Ulman, J., *The Design and Analysis of Computational Algorithms*, Mir: Moscow, p. 535, 1979.
[10] Traub, J. & Voznyakovskiy, H., *The General Theory of Optimal Algorithms*, Mir: Moscow, p. 381, 1983.
[11] Fuller, B. & Reyzin L., Computational Entropy and Information Leakage (2011), available at http://www.cs.bu.edu/fac/reyzin
[12] Vatolin, U.N., Information criterion of stability. *Numerical Methods of Mechanics of the Continuous Environment*, **2(3)**, 1971.
[13] Vatolin, U.N., On the application of the entropy bounds of stability, numerical methods of continuum mechanics. *Numerical Methods of Mechanics of the Continuous Environment*, **5(2)**, pp. 5–6, 1974.
[14] Haykin, S., *Neural Networks*, Williams: Moscow, pp. 89–340, 2006.
[15] Kovalevskiy, S.V. & Reshetnik, N.A., Features of systems diagnostics with application of a principle of an entropy minimum. *Proc. of the VII All-Russia Conf. On Neuro computers and their application*, Moscow, pp. 394–396, 2001.