Ministry of Education and Science of the Republic of Kazakhstan
Suleyman Demirel University

Adilet Duman

PLANNING THE UNIVERSITY TIMETABLE USING NEURAL NETWORKS

THESIS

Presented in Partial Fulfillment for the
Degree of Master of Science in Mathematics
(degree code: 6M060100)

Department of Mathematics and Natural Sciences
Faculty of Engineering and Natural Sciences

Supervisor: **Aziza Aipenova**

Kaskelen, 2019

Abstract

Planning the timetable issues with constraints for the most part have a place with the enormous class of NP complete issues. A significant gathering of such issues concerns Timetable Scheduling (TS) for individuals and devices. In this process, we will optimize the algorithm of the lesson schedule at KUU (renamed the university in accordance with administration requirements). However, the planning and settlement of this issue made it difficult for many of these constraints. This work proposes an effective way to deal with TS utilizingq Lateral Inhibitory Networks (LIN) for imperatives the board. TS can be considered as a booking issue with disjunctive limitations, and the Neural Network model with LIN is all around adjusted to it. This kind of neural models and the arrangement picked to take care of the TS issue is depicted. An application, written by Python language utilizing a PC, has been actualized to take care of the TS issue. This way of solving the problem gives a new impetus to the graphics.

Аңдатпа

Сабақ кестесін жоспарлау, алгоритмін құру -- оған қойылатын шарттардың көптігіне байланысты NP толық мәселелерінің үлкен тобына жатады. Бұндай мәселенің маңызды топтарының бірі адамдар мен құрылғыларға арналған Кесте Жоспарлау (TS) түрі. Осы жұмыс барысында KUU (әкімшіліктің талаптарына сәйкес университет аты өзгертіліп алынды) деген университет сабақ кестесінің алгоритмін барынша оңтайландыратын боламыз. Алайда, бұл мәселені жоспарлау және шешу осыған қойылатын шарттардың көптігіне байланысты қиындық тудырды. Осы жұмыста Жанамалы Ингибиторлық Желіліерінің (LIN) тиімді тәсілін TS-да қолданылды. TS дискрюнктивтік шектеулермен жоспарлау мәселесі ретінде қаралып және LIN жүйесіндегі нейрондық желілік моделді оған бейімдедік. TS мәселесін шешу үшін Python (арасында C, Excel) бағдарламалау тілі қолданылды. Мәселені шешудің бұндай жолы сабақ кестесін құрауда жаңа серпін береді.

Аннотация

Планирование расписания – является частью большой группы полноценных проблем в NP из-за большого количества ограничения. Одним из наиболее важных вопросов является тип Планирования Расписания (TS) для людей и устройств. В этом процессе мы оптимизируем алгоритм расписания занятий в KUU (в соответствии с требованиями администрации имя университета было изменено). Тем не менее, планирование и решение проблемы, которая вызвала проблемы с большим количеством требований. В этой работе наиболее эффективный метод Боковые Ингибирующие Сети (LIN) был использован в TS. TS рассматривался как проблема дискретного планирования границ и адаптировался к модели нейронной сети в LIN. Python (частично C, Excel) язык программирования был использован для решения проблемы TS. Такой способ решения проблемы дает новый импульс графику.

1. Acknowledgements

I want to thank all teachers of this faculty for their hard work and patience, for giving us excellent knowledge and giving a start in life, special thanks.

I express my deep gratitude for the help in the preparation of the thesis work for my supervisor - Aziza Aipenova. For her serious scientific approach and strict mastery of science deeply infected and encouraged me. From choosing a topic to the final completion of the work, she always gave me empathetic guidance and tireless support. She helped me a lot in my work, not only helped me correct mistakes in my work, but also gave me valuable advice. I would like to take this opportunity to express my heartfelt thanks and deepest respect.

I thank my comrades and friends for carefully reading this work, expressing their opinions and commenting on my materials.

To all of them and to many others - my sincere appreciation and gratitude.

Contents

2. Introduction

2.1 Motivation

Planning procedures go for helping the arranging and the board of complex different errand ventures. They for the most part attempt to streamline a goal work, as the entire length of a venture or its expense, by fulfilling a few limitations. On account of limitations which just concern the assignments' progression in time, the planning issue is very basic and a few polynomial calculations permit the arranging of broad tasks. Nonetheless, for the situation where planning requirements meddle with designation imperatives because of the restricted accessibility of assets, the issue turns out to be significantly more troublesome and for the most part has a place with the enormous class of NP-complete issues for which a general deterministic polynomial calculation is obscure. At the point when two undertakings or more can't be checked out at the same time, for example if these assignments need the selective utilization of one bit of gear, we talk about planning with 'disjunctive' or fundamentally unrelated imperatives. The complex combinatorial issues with synchronous planning and assignment requirements are amazingly shifted. A significant gathering of such issues concerns Timetable Scheduling(TS) for work force/staff and instruments in manufacturing plants and organizations. For instance, drivers and vehicles in a transportation organization, speakers and gathering rooms during a congress, instructors, learners and homerooms during tests, and again for laborers and machines in a workshop. Various requirements and particularly disjunctive ones must be fulfilled for the TS. TS issues have been considered for quite a while, utilizing operational research methods and chart hypothesis, computerized reasoning, and neural system models, particularly the Hopfield arrange and collaboration rivalry systems.

The normal way to deal with these examinations comprises in coding, if conceivable, every one of the imperatives and, by and large, in inquiring about the ideal (or imperfect) timetable. Up to now, because of the incredible number of limitations, the multifaceted nature of the issue has prompted huge scale applications on supercomputers as opposed to little PCs.

2.2 Aims and Objectives

In this work, school timetable issues are all the more especially contemplated for a few reasons. These timetables are different. They rely upon instructive frameworks in every nation and advance quickly with the instructional method. As the quantity of learners develops, it is important to streamline the utilization of the assets, for example study halls and educators. These perplexing timetables apply likewise to various modern applications as those refered to in the presentation. We have considered a class of timetables, as often as possible experienced in French colleges, which is perplexing (attributable to various requirements) and very broad. This class of timetables is very much shown by the case of a designing estaLINshment. For this situation, the examinations last commonly three school years. For each school year, there are:

- 'courses' which address all learners (for instance 100 learners),

- 'works out' for which the learners are partitioned into gatherings (up to 24),

- 'down to earth work' which concerns similar gatherings or subgroups (up to 12 learners in each).

Activities and commonsense work are rehashed the same number of times as essential. The traditional and disentangled course timetable issue comprises in keeping away from the contentions where courses occurring all the while include basic learners, educators or require similar homerooms. It is an asset task issue. For genuine school timetable applications, the issue turns out to be progressively troublesome since we need to consider extra imperatives:

- the length of courses might be different,

- a few courses, activities or commonsense work should (or should not) be booked during that day. For example, a few courses must be rehashed a few times during the week and appropriated in the timetable. In actuality, some extraordinary courses must be trailed by activities,

- the educators might be inaccessible (or may want to instruct) during certain pieces of the week or of the year. The educators may likewise want to be engaged with back to back timeframes (timetables compacting for instructors). Similarly, study hall accessibility may change during the educating year,

- another troublesome issue concerns the preassignments: a few courses need to happen at fixed periods, as well as in unique study halls.

Clearly, the quantity of limitations increments quickly with the quantity of learners, and after that of showing subjects, study halls and educators. The above mentioned imperatives can be isolated in two classes: hard limitations which should dependably be fulfilled and delicate requirements which are less prohibitive and may be transgressed. Hard limitations, which are basic, are various (around 10 000 imperatives for a building establishment with a few many learners). There exists an extraordinary number of arrangements fulfilling these requirements. Despite the fact that the undertaking of finding a specific arrangement does not require any ability, it is all in all a critical errand for an individual. Delicate imperatives, which are basically inclinations, are a few times mediocre in number to hard limitations. They can be fulfilled either by a human master utilizing his insight into the educational module or by an orderly streamlining procedure which approximates the most ideal the arrangement of principles planned by the human master.

The greater part of the methodologies join the two sorts of requirements in a solitary streamlining issue. As we would like to think, the fulfillment of the hard requirements is a key yet subordinate undertaking, which needs to occur on a lower level than the delicate imperative administration level. The issue of the delicate limitation the board can be treated by various techniques (operational research, man-made consciousness, neural networks,...). Its programmed goals comprises in the interpretation of a lot of guidelines, connected to a learning base of a human master into the type of a streamlining calculation. As a rule, the coding of wishes and delicate necessities is a long errand, hard to characterize and difficult to be done totally, so more often than not, the target capacity to improve isn't well-characterized. In a first time, we concentrate on the goals of the hard limitations issue, realizing that it is a low dimension task which might be called at any minute by a more elevated amount task, executed either by a human master or by a specialist framework, or again by an advancement calculation. For the occasion, it is a human master who executes this assignment. Internationally, the PC deals with the hard imperatives, and the planner builds timetables thinking about the delicate requirements in an intelligent manner. All the more decisively, we have picked:

- to express and to code the hard requirements, managing assets to be shared (homerooms, understudy gatherings and educators), as disjunctive imperatives (a solitary asset can't be chosen all the while by two clients). These disjunctive imperatives are presented effectively and successfully in a neural system with Lateral Inhibitory Networks (LIN), - to enable the architect to build, well ordered, a timetable in an intelligent manner. The yields of the neural system giving him the rundown of the free assets during the TS, for each schedule vacancy, as indicated by his past decisions. In a moment stage, the originator can consider the delicate imperatives without explaining and coding them. Figure 1 delineates our general way to deal with the TS issue.

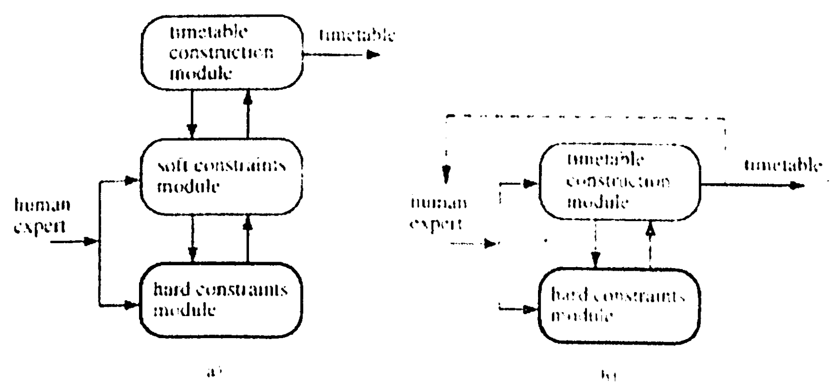


Figure 1. (an) Our way to deal with the TS issue: first the human master gives hard and delicate limitations to the comparing the executives modules, at that point the framework conveys a timetable; (b) the present acknowledgment: first the human master gives hard conditions, and afterward develops a timetable in an intuitive manner as indicated by his very own mastery (the hard imperative module is the equivalent in the two cases).

3. HISTORY OF NEURAL NETWORKS

3.1 Genetic algorithms

The modern bibliography on genetic algorithms has long exceeded 90 titles and continues to increase continuously. However, despite the abundance of literature, it is rather difficult to articulate what they are - the quintessence of evolutionary rearrangements in natural populations of organisms, a universal means of describing adaptations in populations of artificial objects, or a powerful search procedure with global optimization.

This section begins with a comparison of two remarks by J. Holland regarding the adaptation and optimization he made in the prefaces to the first and second editions of his famous book [1], which initiated the process of disseminating genetic algorithms in scientific communities.

True, they began to be called genetic later, and in 1975 Holland called them reproductive plans and considered them primarily as adaptation algorithms. But the shift in emphasis in the interpretation of the concept of adaptation, which he casually says in the preface of 1992, very precisely, conveys the state of confusion that we feel today, trying, on the one hand, to give a fairly general and consistent definition of adaptation and optimization, adaptation and evolution, adaptation and learning.

In the further presentation of the basic ideas of the GA, we will not adhere to the style of the book of Holland, but approach them as a procedure of global optimization. This, although somewhat simplified in comparison with the Holland, tactics of the GA caused a strong resonance in the literature, and as time showed, it is quite reasonable. By and large, almost two decades of GA research of non-test multi-extremal functions went to the proof of this particular facet of GA power, leaving their outstanding adaptive abilities in a certain shadow.

So, GAs are based on the theoretical achievements of the synthetic theory of evolution, which takes into account the microbiological mechanisms of inheritance of characters in natural and artificial populations of organisms, as well as on the experience accumulated by mankind in animal and plant breeding.

The methodological basis of the GA is based on the selection hypothesis, which in its most general form can be formulated as follows: the higher the fitness, will be expressed even stronger. Since HAs deal with populations of constant numbers, elimination selection is of particular relevance along with selection for parents. An elimination strategy designed to answer the question "What individuals can we safely refuse?" is no less important component of modern GA, than the strategy of selection in the parent group. Most often, individuals with low adaptability are not only not involved in the generation of a new generation, but are eliminated from the population at the current discrete evolutionary step.

However, this is true not only for the GA, but for any numerical optimization method. The very idea of optimality, as correctly noted at, came to science from biology. However, we don't always give ourselves the idea of how many methodical techniques of optimal design have roots in practice and are an example of our not always conscious submission to Nature.

Figure 2. Duality of selection tasks, due to the limited size of the population being modeled

It is not difficult to be convinced of the validity of what has been said if we try to look at the procedure of numerical optimization through the prism of the selection hypothesis (see figure 2).

So, usually the design begins with the formation in the search space of the region of permissible values of variables and the choice of some test points in it.

Next, iteratively perform the following steps. First, using a mathematical model, devices produce a mapping of points from the search space to the criterion space, which allows us to get an idea of the surface relief of the criteria. Then, based on the information received and in accordance with the coordinates of points in the variable space, culminating in the generation of coordinates of new test points.

Outlining this familiar chain in general terms, let us note the clearly expressed parallelism between the search for extremum ideology embedded in it and how Nature's similar tasks in nature are solved in adapting populations of organisms to environmental factors.

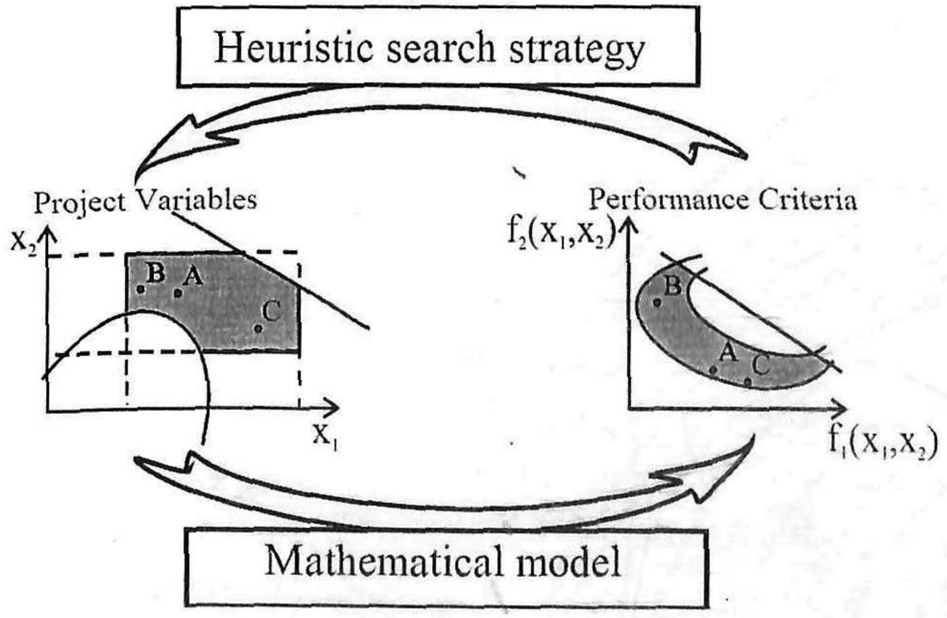


Figure 3. Cyclic structure of the project numerical optimization procedure

Indeed, the practiced way of describing technical objects using design design vectors involves symbolic encoding of information about the object. The vector of variables is not even a drawing, that is, looking at it and without knowing the coding rules, it is impossible to get an idea about the object.

In a sense, it can be argued that the category "vector of design variables" plays the same role in technology as the category of "genotype" in biology. By grouping the key parameters of the object into a vector of variables, we essentially give them the status of genetic information. It is genetic, because, on the one hand, it is enough to build the object itself (hypothetically - to grow it), and secondly, it serves as the source material for generating the next-generation genotypes.

But this is precisely the meaning - the genotypes of descendants - have the coordinates of the new test points mentioned. Just as in Nature, the crossing of organisms is carried out at the genetic level, in the optimization procedure, the coordinates of new test points are obtained as a result of manipulating the coordinates of old ones. Moreover, the selection hypothesis is also invisibly present - the best phenotypically always act as parental ones, not arbitrary points (individuals) from the population of potential solutions, but unsuccessful solutions are discarded at the current step (we can assume that they are dying out).

Here we come, finally, to what exactly distinguishes the GA against the background of other numerical optimization methods.

GA borrow from biology:

· conceptual apparatus;

· the idea of a collective search for extremum with the help of a population of individuals;

· methods of presenting genetic information;

· methods of transmitting genetic information in a series of generations (genetic operators);

· the idea of the preferential reproduction of the fittest individuals (it is not about whether the given individual will give offspring, but about how many descendants it will have).

3.1.1 Presentation of genetic information

Just as the natural chromosomal material is a linear sequence of various combinations of four nucleotides (A - adenine, C - cytosine, T - thymine and G - guanine), the vector variables in GA are also recorded in the form of chains of symbols, using two, three or four-letter alphabet. For simplicity, consider the case of binary coding used in modeling the evolution of haploid populations.

So, we will assume that each variable x_i is encoded by a specific fragment of the chromosome consisting of a fixed number of genes (see figure 3). All chromosome loci are diallel — that is, in any position of the fragment both zero and one can stand. Nearby fragments are not separated from each other by any markers, however, when decoding chromosomes into a vector of variables, the same encoding mask is used during the entire mediated period of evolution.

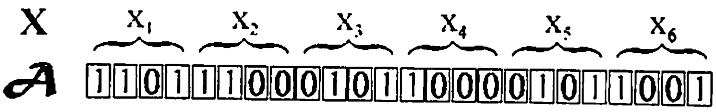


Figure 4. The simplest chromosome mapping mask, defining a plan for the distribution of hereditary information along the chromosome length

Although we are constantly talking about decoding, in fact, the direct operation, understood as the operation of encoding the vector of variables x in chromasome , is not accepted in GA. Chromosomes are generated randomly, by successively filling the digits (genes), immediately in a binary form, and any subsequent changes in the population affect the genetic level first, and only then the phenotypic effects of these changes are analyzed, but never vice versa.

In principle, any binary - decimal code is suitable for decoding genetic information from a binary form, but usually it is assumed that it is represented in the Gray code. Table 1 reproduces in full volume the procedure for decoding a fragment of the chromo-mass into the projection of the vector of variables x_i.

From the gray code we go to the binary - decimal code, and from it - to natural integer numbers. The ratio of the resulting number to the maximum number available for coding a_i given number of bits of the fragment (in Table 1 is the number 15) gives the desired shift

Gray Code	Binary Decimal Code	Decimal Shift Value	Real Coordinate Value
0000	0000	0	a_i
0001	0001	1	$a_i+1(b_i-a_i)/15$
0011	0010	2	$a_i+2(b_i-a_i)/15$
0010	0011	3	$a_i+3(b_i-a_i)/15$
0110	0100	4	$a_i+4(b_i-a_i)/15$
0111	0101	5	$a_i+5(b_i-a_i)/15$
0101	0110	6	$a_i+6(b_i-a_i)/15$
0100	0111	7	$a_i+7(b_i-a_i)/15$
1100	1000	8	$a_i+8(b_i-a_i)/15$
1101	1001	9	$a_i+9(b_i-a_i)/15$
1111	1010	10	$a_i+10(b_i-a_i)/15$
1110	1011	11	$a_i+11(b_i-a_i)/15$
1010	1100	12	$a_i+12(b_i-a_i)/15$
1011	1101	13	$a_i+13(b_i-a_i)/15$
1001	1110	14	$a_i+14(b_i-a_i)/15$
1000	1111	15	b_i

Table 1. Decoding of chromosome fragments in the projection of the vector of variables

value of the variable relative to the left border and the allowable range of its change, normalized to the width of the $b_i - a_i$ range.

The table clearly shows why the Gray code has clear advantages over the binary-decimal code, which, under certain circumstances, creates a kind of dead ends for the search process. As an example, consider any three adjacent rows from Table 1, for example, encoding a shift of 4, 5, and 6 units.

Suppose the chromosome fragments standing in the fifth line and encoding the number 5 belong to the optimal vector, which is the solution of some problem, and the best individual from the current population contains a chromosome fragment from line 4. This situation is favorable for both codes. It is enough to perform only one operation - replace the 0 in the fourth digit of the fragment with 1 - and the solution will be found. A more interesting case is obtained if the best individual contains a fragment from line 6. For the Gray code, the situation is no more complicated than the previous one - replacing 0 with 1 in the third digit will again lead to success. At the same time, the binary-decimal code makes us need to perform two operations in succession - replace 1 with 0 in the third bit and 0 with 1 in the fourth.

From whichever of them we start, the result will not bring us closer to the solution (the first option will replace us in the fourth line, and the second - in general in the seventh line). But this is not the worst example - working with combinations of 3-4, 7-8, 11-12, etc. binary coded decimal strings are even more complicated. In other words, if we draw in geometric interpretations, the Gray code ensures that two neighboring, belonging to one edge, vertices of the hypercube α, which is being searched for, are always decoded into two nearest points of the space of real numbers R^N spaced apart by one precision bit. Binary-decimal code does not have this property.

3.1.2 Genetic Operators

Those mechanisms of transmission of inheritance, which operate in Nature, and the simplified form of which forms the basis of what we call genetic operators, in fact, should be considered as winners who prevailed in a tense, centuries-old battle over competitors and polished by natural selection to the same extent. , like all that surrounds us. Today it is clear that genetic operators could be borrowed not only from microbiological research, but also from the analysis of linguistic phenomena (it is enough to analyze the combinatorial heuristics used by man in solving crossword puzzles) or inventive activity 4. But it is today; and twenty years ago it was necessary to possess the genius of J. Holland in order to figure out how to interpret the principles of the action of "biological" mechanisms for solving problems of adaptation in artificial systems.

Perhaps the main result of almost a quarter of a century-long study of the GA itself was the understanding of the excellent mutual complementarity of the triad of genetic operators "crossover - mutation - inversion". Acting with some probability on the genotypes of the parental individuals, each of them, on the one hand, ensures the transmission of progeny vital signs to the offspring, and on the other hand, maintains a fairly high level of variability throughout the evolutionarily significant period. The whispering in the offspring of new phenotypic characters, different from the parental ones, opens up additional opportunities for the population to adapt, that is, it helps to preserve its search ability.

So, the mutation operator (see figure 5), like point mutations in Nature, is interpreted as replacing the existing allele state of an individual gene in the chromosome with the opposite (idiene - zero and vice versa). Obviously, depending on which bit of the fragment encoding the variable mutates, the size of the distance separating the descendant from the parent (this is not Hamming's space α^L, where this distance is 1, but the space of real numbers R^N).

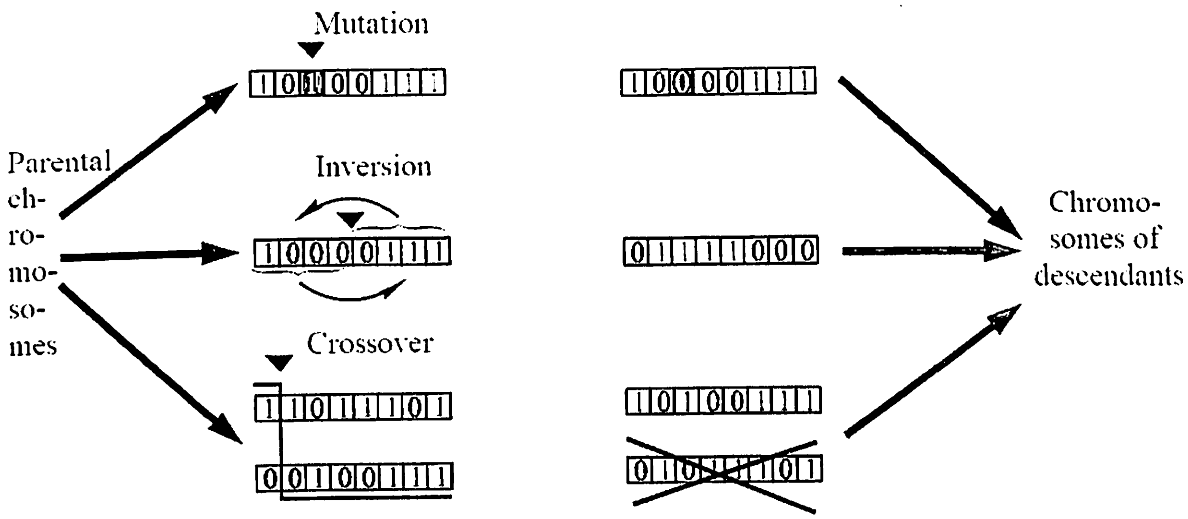


Figure 5. Triad of genetic operators

Inversion leads to a violation of the sequence of chromosome fragments in the descendant compared with the parent chromosome. Finally, a crossover describing the mechanism of gametogenesis in diploid populations of organisms and introduced by Holland into modeling the evolution of haploid populations causes the descendant chromosome to include two fragments, one of which belonged earlier, conditionally speaking, to the paternal chromosome, and the other maternal. It is precisely due to the presence of crossover exchanges that individuals of the population exchange genetic information with each other, that is, the search takes on a truly collective character.

Sometimes, speaking of the triad of genetic operators, it emphasizes the ability of the crossover and inversion to the global search, while the mutation is identified with the means of local adjustment of the solution, giving it a background role. Such a distribution of roles seems controversial, since a mutation can give rise to a descendant far beyond the local extremum in which the parent is located, on the other hand, a crossover conducted on the gametes of the parents located in a common extremum will most likely create offspring in the same extremum. Another thing is important - neither the crossover nor the mutation in the process of generating a descendant is based on knowledge of the local topography of the target function. In this sense, they can be considered global.

3.1.3 Preemptive right to breed the strongest

The thinking style adopted in biology is very different from technical thinking. In biology, the smallest unit, significant in an evolutionary sense and worthy of attention, is the population, not an individual. The fact that the population is adapted to the environment, how well it develops, is judged by the dynamics of its population. It is not so interesting, whether the horns of deer became more branchy, it is important that the increase in the number of the herd be positive. The breeding rate, averaged over the population, is considered as the only and universal criterion for the population's adaptation to habitat conditions [5].

On the other hand, the individual fitness of an individual has a direct impact on the future of the population. The more descendants of this individual will live to reproductive age, the greater the number of members of the population of the future generation will carry its alleles. "Adaptation, considered as a measure of the influence of a genotype on the future," Holland writes in [1], "represents an idea that is useful across the whole spectrum of adaptation problems. A good way to view this idea in a wider context is to consider testing genotypes as a sampling procedure. The sample space in In this case, it is a set of all Alpc genotypes, and the result of evaluating each structure is the fitness of the corresponding phenotype. This is a common question related to fitness, To what extent does an Ain Lambda $\mu_E(A)$ estimate affect or change the Tau design of a new sample? "Looking back rather than before, we are confronted with another interconnected question:" How does the testing history of previous samples affect the current formation plan for new ones? " Answers to these questions go far to determining what constitutes the basis of any adaptive process. "

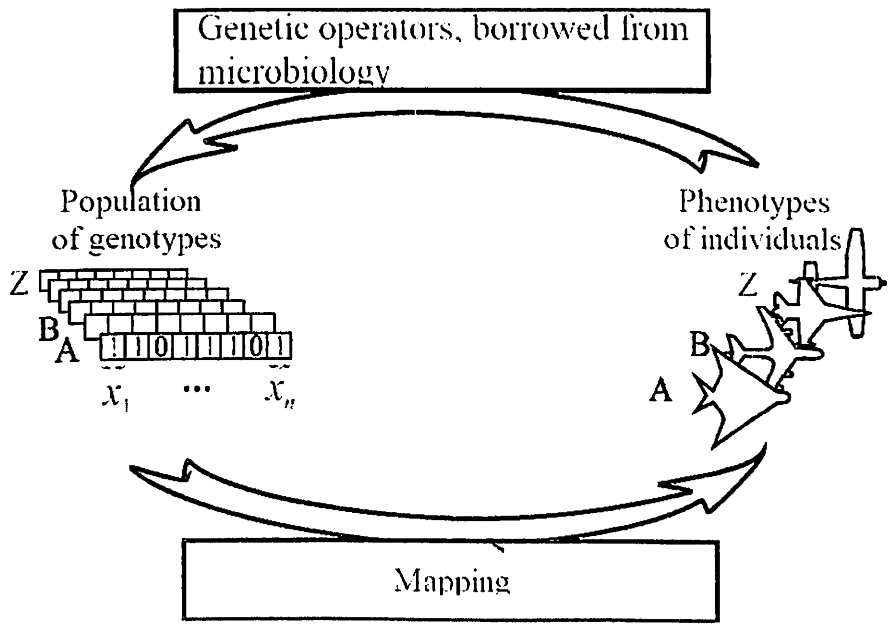


Figure 6. Transformation of hereditary information into GA

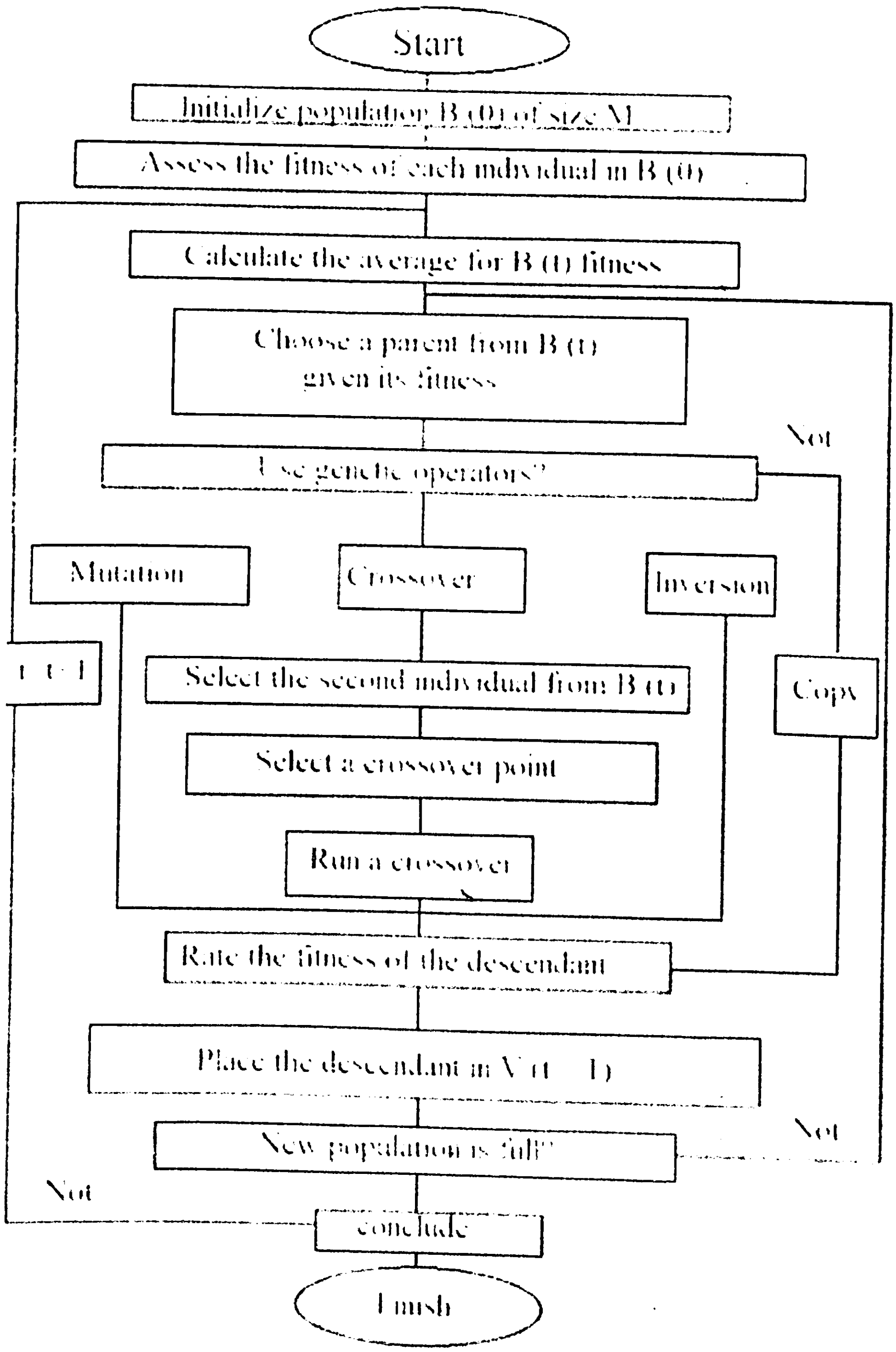


Figure 7. Holland's reproductive plan

3.2 Neural networks

The class of neural systems (NN) is a subclass of parallel appropriated handling models. These models expect that data handling is a consequence of associations between more straightforward preparing components (nodes).

Neural system models comprise of:

1) A set of preparing nodes.

2) A condition of enactment for every node.

3) An example of network among nodes (a diagram).

4) A propagation regulations.

5) An activation rule.

Neural system comprises of various handling components (nodes) associated in a chart. For the most part, every hub speaks to some element of the issue space being investigated. Every hub gets information esteems (from different nodes), keeps up a condition of enactment, and sends yield esteems (to different nodes). Much of the time, the enactment of a hub is some genuine numeric amount (typically extending over a lot of discrete qualities or taking on any genuine incentive inside some range). Nonetheless, some of the time the enactment is just paired. A standard of proliferation decides how the yields from nodes are consolidated to frame contribution for different nodes. For the most part, this is just a weighted aggregate in which the associations between nodes are doled out loads. Positive loads can speak to excitatory associations, and negative loads inhibitory associations. At last, every hub has an actuation rule which decides another condition of enactment for the hub, given a lot of contributions to that hub and its present condition of initiation. There exists a huge assortment of neural systems. This work focuses just on those that have been utilized regularly to take care of combinatorial advancement issues: imperative fulfillment systems. This segment is isolated into three subsections.

3.2.1 How it works?

The network of neurons that forms the human brain is a highly efficient, complex, substantially parallel information processing system. It is able to organize its neurons in such a way as to realize the perception of the image, its recognition is many times faster than these tasks will be solved with the most modern computers. So the recognition of a familiar face occurs in the human brain for 100 - 200 ms, while the computer needs minutes and even hours for this.

Today, as 40 years ago, it is certain that the brain works more efficiently and in a fundamentally different way than any computer created by man. It is this fact that for so many years prompts and guides the work of scientists in the creation and research of artificial neural networks.

3.2.2 Stepwise development of nervous system

The question of whether a living organism is something like a machine was first raised by the French philosopher and mathematician Rene Descartes (1596-1650). Descartes lived during the rise of mechanics, when Kepler and Galileo began to develop ideas about the movement of celestial bodies. Radically started to develop ideas about the movement of celestial bodies. Radically new views on man and the universe then received only the first impetus. Prior to this, ideas had reigned that the laws of Nature, beginning with the fall of the stones and ending with the movement of the planets, are immutable and unshakable. The Universe was represented as a clock mechanism striking the imagination with its scale, created and set in motion by the Great Creator. At the household level, these ideas were embodied in the mechanical trinkets created for the inhabitants of wealthy European houses. A cuckoo clock, fountains in the alleys, doused by visitors who accidentally stepped on a hidden spring — could it not be possible to explain the thoughts and actions of a person in similar mechanistic terms?

According to Descartes, all actions, both in man and in animals, are a response to the events of the outside world. An external stimulus excites one of the senses. Excitement is transmitted to the brain, which directs it to the appropriate muscle, that is, to develop a reaction to an external stimulus. The energy of the stimulus is reflected back by the nervous system through the muscles of the animal.

Progress in understanding the mechanisms of such reactions was made some- what later on experiments with animals from which the brain was removed. Around 1750, Scottish scientist Witt proved that such movements are controlled by the spinal cord. He found that the decapitated frog pulls the legs away from the pin, but if you remove the head and spinal cord from it at the same time, it stops responding to the injections.

A further push in the study of unconditioned reflexes was obtained during the French Revolution. Pierre Kabani, a friend and physician of some of its leaders, investigated whether the creature persists after guillotine. He concluded that it was not, and that the convulsions of the decapitated body were more like reflex actions. These dark studies were continued by the German researcher Theodor Bischof, who performed a series of experiments on the heads of executed criminals. Even sufficiently strong pathogens did not produce any effect during the first minute of decapitation.

Classical conditioned reflexes were first described at the beginning of the twen- tieth century by I. P. Pavlov, who immediately perceived in them the simplest form of training, thanks to which two events are associated. In the classical conditioned reflex, the initially ineffective stimulus, called the conditioned, is re-combined with the highly effective stimulus, called the unconditioned. Initially, the conditioned stimulus causes only a weak response, or none at all; an unconditioned stimulus provokes a violent reaction without any prior training. As a result of the development of the conditioned reflex, the conditioned stimulus acquires the ability to cause either a strong or a new response. In order to form a conditioned connection, that is, learning has occurred, the conditioned stimulus must correlate with the unconditioned one, preceding it by a certain critical period of creme.

Prior to the beginning of this century, most neurophysiologistsi believe that the route of the reflex runs through essentially continuous filaments of nervous tissue. The concept of synapse, the gap between neurons, through which they must interact, is relatively new. Studies establishing the presence of the synapse and its role in nervous activity were performed at the turn of the century by the English physiologist Sir Charles Sherrington (1857-1952). Sherrington's work involved a behavioral level, not an electrophysiological one. Nevertheless, from the analysis of the reflex actions of dogs, cats and obyazyan he managed to unravel the basic principles of the synapse.

The first attempts to unlock the secrets of the anatomical organization of the brain include research by Santiago Ramon y Cajal (1911). Applying the method of coloring neurons with silver salts, developed earlier by Camillo Golgi (silver selectively penetrates neurons but does not infiltrate other brain cells), Kahal saw that the brain has a cellular architecture. In addition to neurons, the structure of the brain includes various glial cells that perform supporting functions and participating in repair processes. Kahal described neurons as polarized cells that receive signals from highly branched processes, called dendrites, and send information to unbranched processes called axons (see Figure 7).

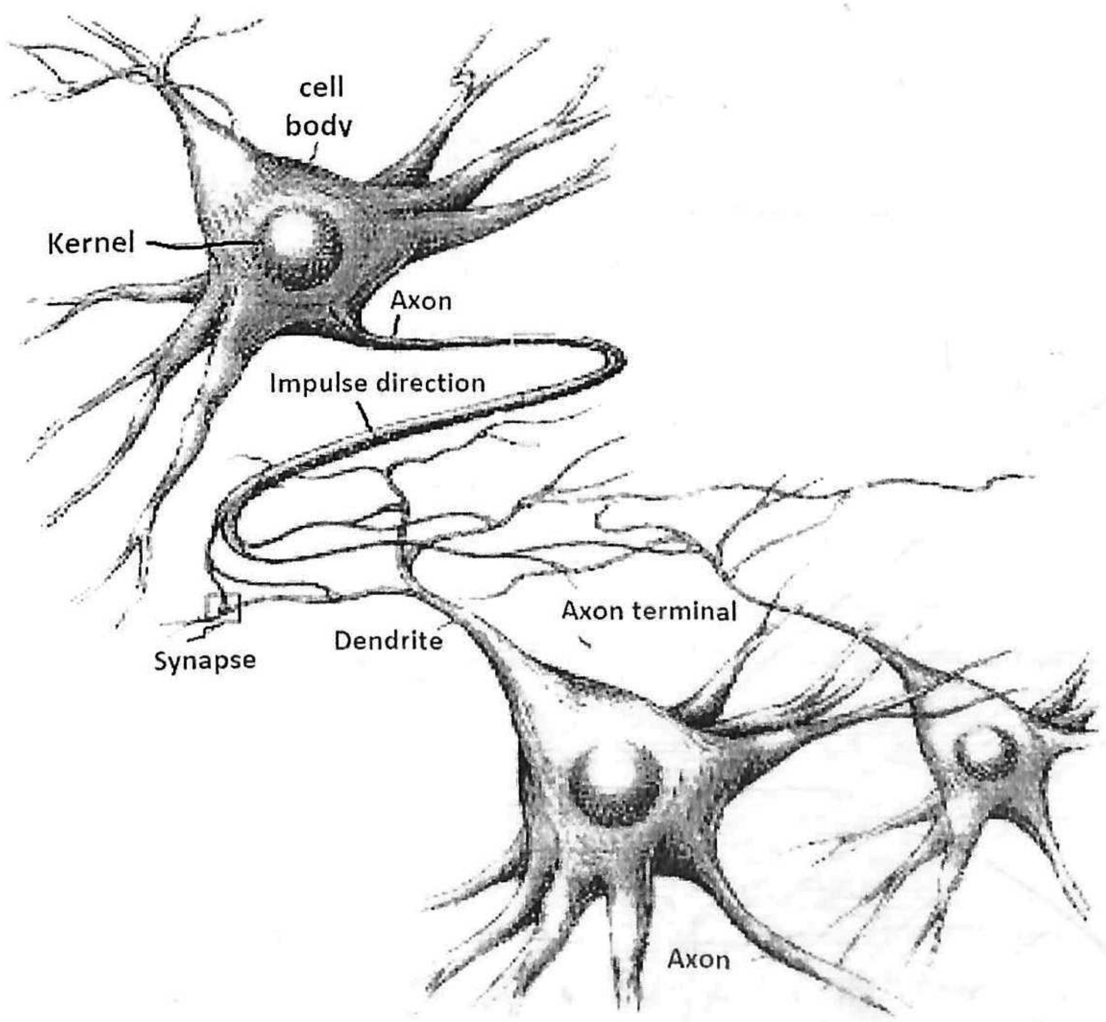


Figure 8. Ensemble of human nerve cells

Golgi staining revealed a huge variety of neurons in body shape, branching of the dendritic part and axon length. Kahal revealed differences between cells with short axons interacting with neighboring neurons, and cells with long axons projecting to other parts of the brain.

Despite differences in structure, all neurons conduct information in the same way. The information on the axions is in the form of short electrical impulses, the so-called action potentials, the amplitude of which is about 100 mV and the duration is 1 ms. The emergence of pulses is associated with the movement of positively charged sodium ions through the surface cell membrane from the extracellular fluid into the cell, into its cytoplasm.

Concentration in the extracellular space is approximately 10 times its intracel-lular concentration. At rest, a transmembrane potential difference of about -70 mV is maintained. At the same time, sodium ions penetrate the cell slowly, since access there for them is limited by membrane properties. Physical or chemical stimulation depolarizing mebran increases its permeability to sodium ions. The flow of sodium into the cell depolarizes the membrane even more, making it more and more permeable.

When a certain critical value of potential, called a threshold, is reached, positive feedback leads to regenerative shifts, as a result of which the sign of the potential difference changes to the opposite, that is, the internal content of the cell becomes positively charged in relation to the external environment. After approximately 1 ms, the permeability of the membrane for sodium falls and the transmembrane potential returns to its value at rest -70 mV. After each such explosion, the neuron remains refractory for a few milliseconds, that is, the sodium permeability of the membrane during this period cannot change. This puts a limit on the frequency of pulse generation - 200 times per second.

Although the axons are similar to the wires, they conduct 6 impulses differently. Their cable characteristics are unimportant: the resistance along the axis is too large, and the membrane resistance is too low. Positive charge dissipates after 1-2 mm. To overcome the distance, sometimes several centimeters, the impulses must be regenerated. The need to re-amplify the current limits the maximum propagation velocity of the amplified current limits the maximum propagation velocity of the nerve impulse along the axon to 100 m / s.

Connections between neurons are mediated by chemical transmitters - neuro- transmitters - that stand out from the endings of the processes of neurons in synapses. When the action potential reaches the end of the axon, the mediator molecules leave the intracellular small vesicles, where they are stored, in the synaptic cleft - a 20 nm wide space between the membranes of the presynaptic and postsynaptic cells. When the excitation reaches a peak, a coordinated release of the neurotransmitter molecules begins.

The released neurotransmitter molecules bind to receptors in the postsynaptic membrane, which changes its pronability. The effect will be exciting if the charge change brings the membrane potential closer to the threshold of pulse generation. If the membrane is stabilized at the level of resting potential, the effect will be inhibitory.

Each synapse has only a minor effect on the activity of the neuron axon. To establish the output intensity, each neuron must continuously integrate up to 1000 synaptic inputs.

At the beginning of the century, the extremely important role of synapses in learning became clear to neurophysiologists. Signals of the brain, passing through them, can be amplified to varying degrees, but weakened. This fact attracts attention. The brain of a newborn and the brain of an adult contain about the same number of neurons. But only the brain of an adult is distinguished by the orderliness of the interneuronal synaptic connections. Apparently, brain training is a process of changing the architecture of a neural network, accompanied by setting up snaps.

3.2.3 New level of neural networks

The most concise is the following definition of the ANN as an adaptive machine, given in: An **artificial neural network** is a substantially parallel distributed processor that is capable of preserving and representing experienced knowledge. It is similar to the brain in two aspects:

1. Knowledge is acquired by the network in the learning process;

2. To preserve knowledge, the forces of interneuron connections, also called synaptic weights, are used.

The history of the NN begins in 1943, when McCulloch and Pitts proposed a model of a "threshold logical neuron" and showed that any function that can be calculated on an electronic computer can also be calculated by a network of neurons. The signals x_i, which enter the neuron input, are multiplied by the weighting coefficients w_i (synaptic weights).Then they are summed, the resulting signal is shifted by the offset w_0.

$$S = \sum_{i=1}^{n} w_i x_i + w_0, \tag{1}$$

is fed to the input of the block that implements the activation function of the neuron.

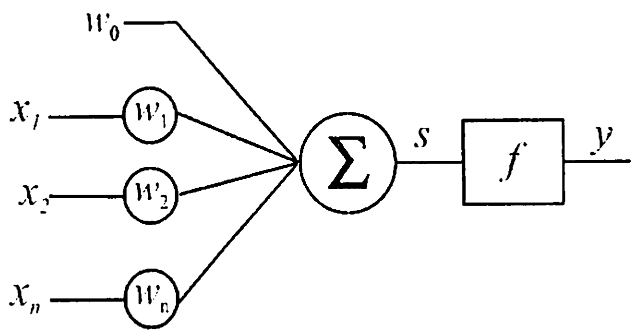
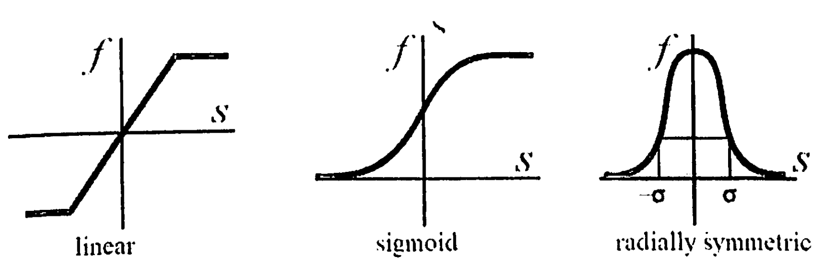


Figure 9. Neuron model

Traditionally, the activation function has a stepwise form, that is, the signal at the output of the neuron y appears only when the total input action exceeds a certain critical value.



$$f(s)=\begin{cases} a, \text{ если } s>a \\ s, \text{ если } -a \le s \le a \\ -a, \text{ если } s<-a \end{cases} \qquad f(s)=\frac{1}{1+e^{-s}} \qquad f(s)=e^{-\frac{s}{\sigma}}$$

Figure 10. Types of activation functions of neurons

Studying the cellular mechanisms of brain activity, Hebb [8] formulated a learning rule that increases the strength of the connection between pre-and post-synaptic neurons, if the activity of both coincides in terms of efficiency.

Another concept of training within the framework of a more developed network architecture, called a percentron, was proposed and successfully applied to simulate the work of the optic tract by Rosenblatt [9].

In its simplest version, the multilayer perceptron (see Figure 11) is a network with one input, one output and one or more internal or, as they say, hidden layers of neurons. A common feature for all multilayer perceptrons is the direct direction of the network, characterized by the transfer of information standard topology, the node i layer k, $(k = 1, ..., k + 1)$ is connected by means of weights w_{ij}^k with all j nodes of the previous layer $k - 1$. Here $k = 0$ and $k = K + 1$ denote the input and output layers respectively.

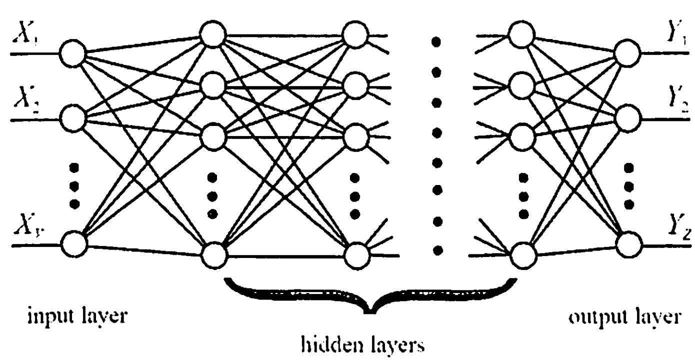


Figure 11. Diagram of the direct directional layer ANN

Modified versions can have direct links between non-adjacent layers, links within one layer, chaotic links between layers instead of regular ones.

The input layer of the perceptron serves only to receive and retransmit input signals to the neurons of the hidden layer. In the hidden layers, the main non-linear transformation of information occurs, and the output layer superimposes the weighted signals of the last of the hidden layers. As nonlinearity, nodes of the hidden layer use differentiated sigmoidal functions

$$f(S) = \frac{1}{1+e^{-S}} \qquad (2)$$

Perceptron training is the purposeful process of changing the weights of interlayer synaptic connections, iteratively repeated until the network acquires the necessary properties. The basis of training is the use of training data, combined in templates (see figure 12).

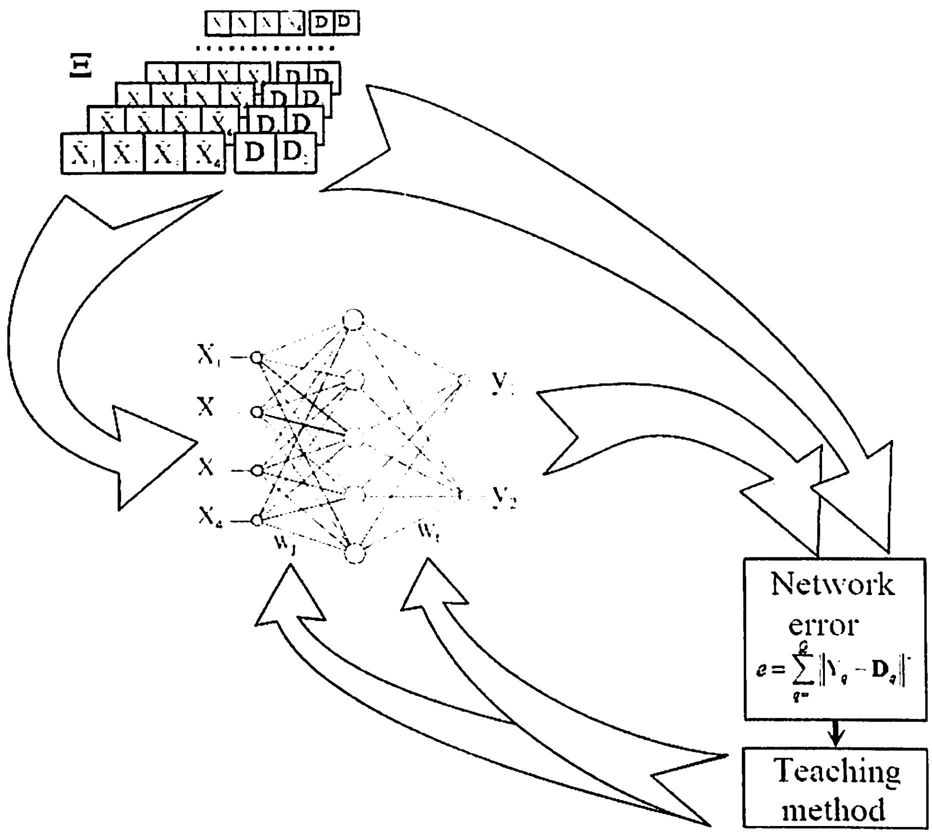


Figure 12. Controlled learning ANN

Each pattern $\langle \breve{X}, D\rangle$ includes a vector of known input signals of the $\breve{X}= (\breve{X}_1, \breve{X}_2, ..., \breve{X}_V)$ network and a corresponding vector of the desired output signals $D = (D_1, D_2, ..., D_Z)$.In the course of training, data from the training set of patterns $\{\langle \breve{X}, D\rangle_q, q = 1, \bar{Q}\}$ are successively dropped into the ANN input, after which the error between the actual $Y = (Y_1, Y_2, ..., Y_Z)$ and the desired outputs of the network

$$e = ||Y_q - D_q||^2 \tag{3}$$

is calculated.

Here, the norm $||\cdot||$ is usually understood as the Euclidean distance between the vectors Y and D.

Further, with the help of a certain rule or algorithm, such modification of the network configuration parameters occurs so that this error decreases. The process is repeated until the network achieves the ability to perform the desired type of "input-output" transformation, specified implicitly by a training set of patterns.

Thanks to the training, the network acquires the ability to respond correctly not only to the templates presented in the training process, but also to cope well with other data sets from the allowable input space that she has never "seen" before. In this sense, they say that the NN has the property of generalization.

Errors in the generalization, always taking place at the output of the network, has two components. The first of these is due to the insufficient quality of the approximation performed by a network of finite size. The second is caused by the incompleteness of information provided by the network in the learning process, due to the limited size of the training sample.

In Rosenblatt, the strength of the interlayer synaptic connections varied depending on how accurately the output of the perceptron coincided with the output template, in accordance with the following learning rule. The weights of connections increase if the output signal generated by the receiving neuron is too weak and decreases if it is too high. However, this simple rule of minimizing the error applies only to direct-directed networks without hidden layers.

Somewhat later, Minsky and Papert performed a deep analysis of the computational power of a single-layer perceptron.

Error propagation algorithm.

Step 1. Initialization of weights and offsets.

$w_{ij}^{(k)}$ weights and $\omega_{i0}^{(k)}$ offsets in all layers are randomly set as small values, for example, in the range from -1 to +1.

Step 2. Presentation of the new input vector $\breve{X}$ and the corresponding desired output vector D.

Step 3. Direct passage: calculation of the actual output. Calculation of the output $Y_i^{(k)}$ for the i-th node in the k hidden layer Y_i:

$$Y_i^{(k)} = f_\sigma\left(w_{i0}^{(k)} + \sum_{j=1}^{H_k} w_{ij}^{(k)} Y_j^{(k-1)} \right) (k = 1, \dots, K). \text{ где } Y_j^{(0)} = X_j.$$

$$Y_i = f_\sigma\left(w_{i0} + \sum_{j=1}^{H_K} w_{ij}^{(K+1)} Y_j^{(K)} \right). \qquad (4)$$

Here H_k- is the number of nodes in the k-th hidden layer.

Step 4. Return pass: adaptation of scales and thresholds.

Using a recursive algorithm starting at the output nodes and returning to the first hidden layer:

$$w_{ij}^{(k)}(t+1) = w_{ij}^{(k)}(t) + \eta\delta_i^{(k)} Y_j^{(k-1)} (k = 1,\ldots,K+1) \qquad (5)$$

For $k = K + 1$ the σ_i^k term describing the error is known:

$$\delta_i^{(K+1)} = (D_i - Y_i) Y_i (1 - Y_i) \qquad (6)$$

and it can be recursively calculated for all other cases:

$$\delta_i^{(k)} = Y_i^{(k)}\left(1 - Y_i^{(k)}\right) \sum_j \delta_i^{(k+1)} w_{ij}^{(k+1)} (k = 1,\ldots,K). \qquad (7)$$

Note that the term $Y_i^k(1 - Y_i^{(k)})$ is a derivative of the sigmoidal function with respect to its argument. If another threshold function is used, this member must be changed. The training parameter T is usually chosen in the range from 0 to +1.

Step 5. Repeat from step 2. Real progress was made only after Rumelhart, Hinton, and Williams in 1986 consistently rediscovered the Inverse Propagation Error (IPA)[11,12] algorithm, first described by Verbos [13] in 1974.

However, with the advent of IPA, interest in neural networks was revived again. It is impossible to ignore the fact that by the end of the 80s the general situation in the world of science had changed significantly compared to the 60s - the progress in the development of personal computers significantly expanded the boundaries of numerical experimentation, the era of numerical simulation methods began. Artificial neural networks become a mass hobby and, through their fans, penetrate the most diverse scientific disciplines.

Consider the procedure for training the RBF network (see Figure 12), which approximates the function specified in a non-manifest form with a set of templates, as described in [16].

Let V - the number of network inputs, H - the number of hidden layers, Z - the number of network outputs.

Suppose that the size Q of the set of training patterns θ is not too large and that the templates are placed rather irritably in the input space of network $X = X_1, X_2, ..., X_V$.

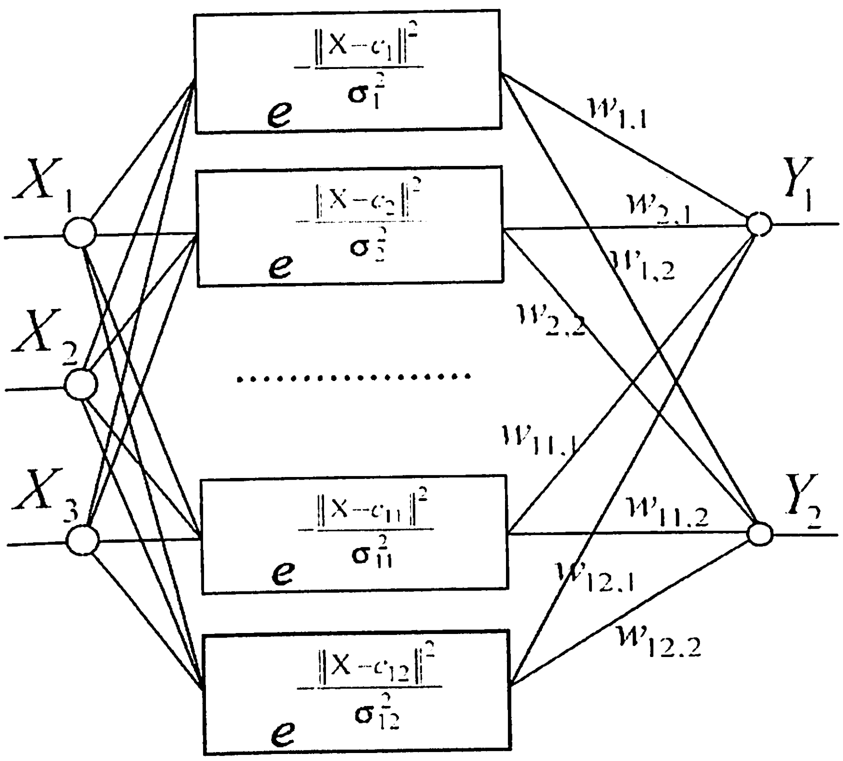


Figure 13. Classic IPF network

We introduce the following notation:

$c = (c_1, c_2, ..., c_V)$–coordinate vector of the activation center function of the neuron of the hidden layer;

δ_j– the width of the window of the activation functions j of the neutron of the hidden layer;

$$f(\mathbf{X}.\mathbf{c}) = e^{-\|X-c\|^2/\sigma^2} = e^{-\sum_{j=1}^{V}(X_j-c_j)^2/\sigma^2} \qquad (8)$$

radially - symmetric activation function of the neuron of the hidden layer; w_{ij}–the weight of the connection between the i neuron of the output layer j and the neuron of the hidden layer.

4. LATERAL INHIBITORY NETWORKS

4.1 One-dimensional general model

This system has been initially intended for sign and picture handling applica- tions We will clarify how it tends to be utilized for our concern. Give us a chance to consider the one-dimensional system appeared in Figure 13 (a). It is a system with N neurons, N information sources and N yields. The associations are nearby and fixed (no learning). An info $e(k)$ is connected to neuron N_k and to its neighbors with fixed loads, diminishing with separation. These are *excitatory associations.* Correspondingly, the yield $s(k)$ of neuron N_k is nourished back to the neighboring neurons with a weight (negative for reasons of security) diminishing with separation. These associations are called Backward Lateral Inhibitions (LIN) . Such inhibitory associations are experienced (frequently by methods for between neurons) in tactile frameworks and in the cerebral or cerebellar cortex. This example of excitatory and inhibitory associations is rehashed with a similar weigh- ting coefficients for every one of the neurons of the system.

The associations are in charge of two fundamental qualities:

- the excitatory associations altogether animate neurons which are in their field of impact, along these lines prompting an impact of collaboration.

- theLateral Inhibitory Networksof one neuron will in general diminishing the movement of its neighborhood: the more a neuron is dynamic, the more the neurons of its neighborhood are hindered. By methods for these associations, the neurons are in rivalry with their neighbors: the neuron (or the gathering of neurons), whose movement is more grounded, will drop that of the area.

The potential $p(k)$ of the neuron N_k is spoken to by the accompanying connec- tion:

$$p(k) = \sum_{i=-\alpha}^{+\alpha} \alpha(i) \cdot e(k-i) - \sum_{j=-b}^{+b} \beta(J) \cdot (k-j) \quad (9)$$

where $\alpha(i)$ (separately $\beta(j)$ are the weighting coefficients of the excitatory (indivi- dually inhibitory) associations, $[-an, a]$ (individually $[-b, b]$) speak to the limits of the area for the excitatory (separately inhibitory) associations. Every coefficient $\alpha(i)$ (or $\beta(j)$) depends just on the separation between the info (or the yield) and the neuron considered. They are autonomous of k. The neuron is mentioned by its potential/ouput trademark appeared in Figure 13. Consequently, the yield $s(k)$ of neuron N_k is:

$$s(k) = min[S_{max}, max[0, A \cdot (p(k) - v)]], \quad (10)$$

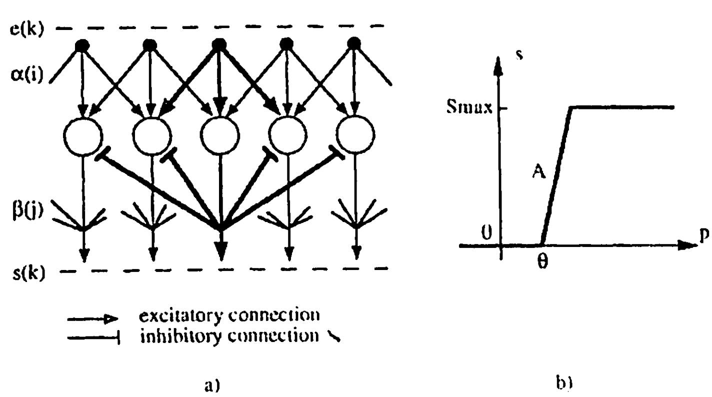


Figure 14. (a) Connection structure of a one-dimensional Network with Lateral Inhibitions Networks(LIN):
(b) transfer characteristic of the neuron

where A is the increase (among 0 and infinity), V is a positive limit and S_max is the most extreme yield esteem. The parameters A, V and S_max are indistinguishab- le for every one of the neurons of the system. Using (9), Eq. (10) moves toward becoming:

$$s(k) = min[S_{max}, max[0, A(\sum_{i=-\alpha}^{+\alpha} \alpha(i) \cdot e(k-i) -$$

$$-\sum_{j=-b}^{+b}\beta(j)\cdot s(k-j)-V)]]. \qquad (11)$$

Such a network, with local connections, maintains the topology. The input and output signals can be considered as spatial ones. From a signal processing point of view, the input connections exhibit a low-pass filtering effect (noise reduction) and the backward lateral inhibitions exhibit a high-pass filtering effect (contrast and dynamic enhancement). Therefore, this network is a non linear spatial filter whose characteristics depend, at the same time, on the neighbourhood size, the connection weights, the gain A, the threshold V and the maximum output value S_{max}. For a network with N neurons, the resolution of the recursive set of N equations (10) is achieved by serial or parallel iterations. This type of network has been used more particularly in image preprocessing for its 'filtering' aspect; we use it here for its 'competition' aspect. In fact, when the inhibition links are strong or when the gain A is infinite, this network exhibits binary outputs $[0, S_{max}]$. The inhibition links appear as 'veto' inhibitions: if a neuron is active, all the neurons connected with it are set to zero. This property is well suited for the coding of disjunctive constraints: if a resource is allocated to one user (active neuron), all other candidates should be inhibited (inactive neurons).

4.2 Adaptation and constrants to coding

4.2.1 Principle

The above mentioned model has been changed so as to code the TS issue and specifically the allotment of assets all the more effectively. By definition, every study hall, every understudy gathering and every instructor is an asset which can be appointed to a course considered as a 'client'. The issue can be reformulated as following:

- a neuron N_k is related to a specific task of a study hall (asset) to a course (client), - an information $e(k)$ licenses the planner to approve (1) or to prohibit (0) such a task (for example, the study hall might be excessively little, or unavailable...) , - the retrogressive horizontal restraints speak to the incongruence of requirements between one study hall (asset) and a few courses (clients), or between one course and a few study halls. In Section 5.1

we will demonstrate that the others sharing asset imperatives (understudy gatherings and educators) lead to present extra LIN, - the yield $s(k)$ shows, as per all the sharing asset limitations, that the task is viably conceivable or unthinkable (1 or 0). At the point when an inhibitory association exists (individually, does not exist), its weight is equivalent to 1 (resp. 0). At that point the yield $s(k)$ of neuron N_k can't be equivalent to 1 (the task is unimaginable) since at any rate one of these inhibitory information sources is dynamic (the asset is inaccessible).

Give us a chance to consider the one-dimensional system, appeared in Figure 14 (a), where each information $e(k)$ is connected to just a single neuron (no dispersion to the neighbors). Inhibitory associations, not only nearby, exist just between neurons which can't be at the same time initiated (for instance: $N_1 - N_2, N_3 - N_4 - N_s$). In the general instance of a system with N neurons, the potential $p(k)$ of the neuron N_k is spoken to now by the accompanying relations:

$$p(k) = e(k) - \sum_{h=1}^{+N} w(k,h) \cdot s(h) \tag{12}$$

$$\text{with } e(k) \in [0,1],$$

$$w(k,h) \in [0,1],$$

$$s(h) \in [0,1],$$

$$\text{and } p(k) \in Z$$

where $e(k)$ is the approval contribution of the neuron N_k, $s(h)$ is the yield of the neuron N_h, and $w(k,h)$ is the inhibitory association weight between the source neuron N_h and the objective one k. For a system with N neurons, the distinctive association loads $w(k,h)$ between neurons can be spoken to by a N by N network W.

$$s(k) = min[1, max[0, e(k) - \sum_{h=1}^{+N} w(k,h) \cdot s(h)]]. \tag{13}$$

The likelihood to incidentally clip to level one the yield of one neuron is presented. This permits the determination of a specific study hall task among all the potential ones. When one neuron is braced, every one of the neurons of the system, which are in shared hindrance with this neuron, become idle.

The yield $s(k)$ of neuron N_k is:

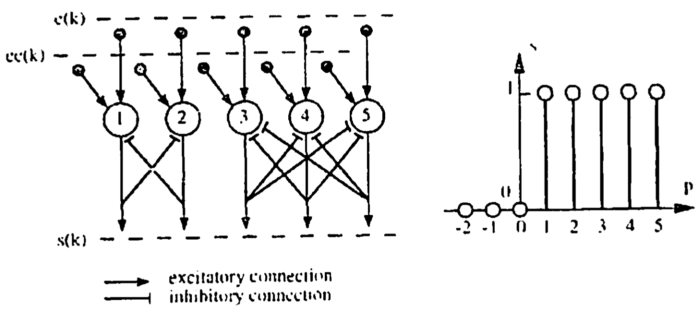


Figure 15.The neuron yield capacity is portrayed by the potential/yield trademark appeared in Fig. 15(b). This capacity is identical to a unit venture with a deferral. The yield $s(k)$ of neuron N_k is at that point:

$$s(k) = min[1, ec(k) + max[0, e(k) - \textstyle\sum_{h=1}^{+N} w(k,h) \cdot s(h)]], \tag{14}$$

where the information $ec(k)$, called bracing info, clips momentarily the yield to level one when it is actuated.

The dynamic framework, spoken to by the neural system, is deterministic and nonlinear. It has a limited number of states: 2^N if there are N neurons. An association grid $W[w(k,h)]$ compares to the system engineering. All the more decisively, it mirrors a picture of the subnetworks which are free and totally interconnected (association loads are equivalent to 1). With parallel (synchronous) refreshing of the yields (goals of the arrangement of conditions), each subnetwork joins as far as possible cycle which is uncertainly secured. The length of the cycle, that is the quantity of progressive states in a cycle, can be equivalent to two or one. For a length equivalent to two, the subnetwork wavers inconclusively between two states. For a length equivalent to one, the subnetwork is steady. The length of the cycle relies upon the actuation condition of the cinching inputs. The system with LIN, that we have depicted, shares a few basic focuses with the Hopfield model: the yield of the neuron is double (0 or 1), the associations are symmetrical. In any case, for our situation, the system isn't totally interconnected and there is no learning calculation: the paired associations (0 or 1) license, only, to retain clashes between neurons related to the distributions of a similar asset. It likewise varies from the Hopfield arrange by the decision of the emphasis mode. Emphasess are processed in a parallel manner, instead of in a sequential one, bringing about a specific dynamic: the completely associated subnetworks may then produce farthest point cycles of length two, which are misused by the planner so as to know, at each minute, which assets are still left free considering naturally his past assignment decisions. Two yield advancement models are displayed beneath for the system appeared in Fig. 15(a). Neurons N_1 and N_2 relate to two potential assignments of a solitary (asset A) between two clients. The asset (A) can be appointed uniquely to one client in the meantime, in this manner the totally unrelated imperatives are accomplished between neurons N_1 and N_2. For a similar reason, neurons N_3, N_4 and N_5 relate to three potential assignments of a solitary (asset B) among three clients. When we have not picked the asset distributions, that will be that no bracing is enacted (Fig. 15(a)), the neuron yields from N_l to N_5 change their states for every cycle due to the totally unrelated imperatives. This implies the two assets are as yet accessible. On the other hand, when we have alloted, for instance, the asset B to the client spoken to by neuron N_3 by physically actuating its bracing info (Fig. 15(b)), the yields of the neurons N_4 and Ns stay at zero. This implies, for this basic precedent, that the asset B is never again accessible for the clients spoken to by neurons N_4 and N_5; henceforth, the product must preclude the enactment of the bracing contribution of neurons N_4 and N_5. Similarly, the asset. A can be relegated to one of the two clients by actuation of the clipping contribution of neuron N_l or N_2. It ought to be noticed that when the clipping information is expelled, the condition of the subnetwork stays stable which implies that the asset has been apportioned to one client.

4.2.2 Improvements

The system with LIN grants a simple coding of fundamentally unrelated imperatives. So as to restrain the expansion of the size of the association grid (N neurons initiate N^2 paired associations), we propose to present for each gathering of neurons in shared prohibition (N_k) an inhibitory neuron NT_h. At that point, the neurons in shared rejection are not associated together (Fig. 14(a)) yet associated with the inhibitory neuron (Fig. 16). The twofold yield (0 or 1) of the inhibitory neuron NT_h goes quickly to 1, when at any rate one of the yields of the neurons N_k (in a similar gathering g) is equivalent to 1. The yield of the neuron NT_h is:

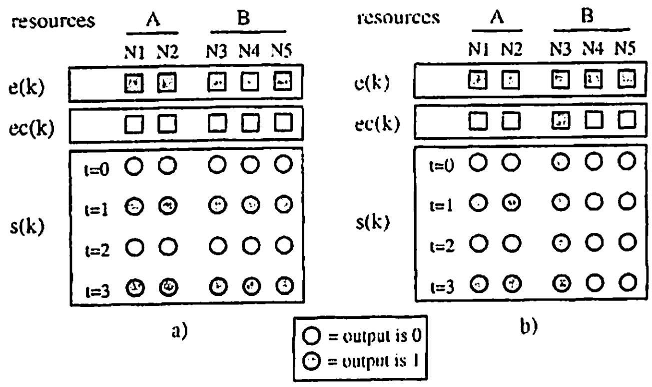


Figure 16. Instances of conduct (for four emphasess) of the system appeared in Fig. 15(a): (a) when no clipping information is enacted, subnetworks merge as far as possible cycle whose length is equivalent to two (first express: all yields are equivalent to zero, second express: all yields are equivalent to one) appearing potential portions, (b) when the bracing contribution of neuron N_3 is initiated, the subnetwork $N_3 \check{} N_4 \check{} N_5$ winds up stable (the length of the point of confinement cycle is equivalent to one relating to the one of a kind conceivable condition of the clasped yield equivalent to every last one the others equivalent to zero).

$$st(h) = min[1, ma[[0, \textstyle\sum_k s(k)]] \tag{15}$$

The yield of the inhibitory neurons is an inhibitory contribution for every one of the neurons in common avoidance. The yield of neuron N_k concluded from (10) is:

$$s(k) = min[1, max[0, ec(k) + e(k) - st(h)] \tag{16}$$

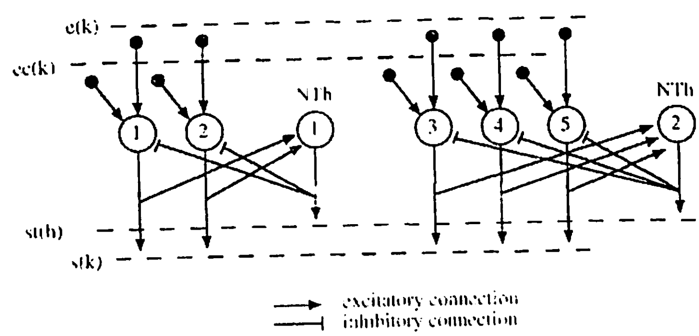


Figure17. The utilitarian conduct of this new system is indistinguishable from the past one. Two yield development instances of the system of Fig. 16 are exhibited in Fig. 17.

5. MODELIZATION OF SCHE ULE CONSTRUCTION PRC LEM

5.1 Allocation constraints

5.1.1 Principle

The asset assignment requirements (hard limitations) are interpreted in the system portrayed in Section 4.2, first, without the improvement of Section 4.2.2 for reasons of lucidity. At that point, the improved arrangement will be given in Section 5.1.2. So as to train the set C of Courses a few arrangements of assets are accessible: R (individually G and T) represents study halls (separately understudy Groups and Teachers).

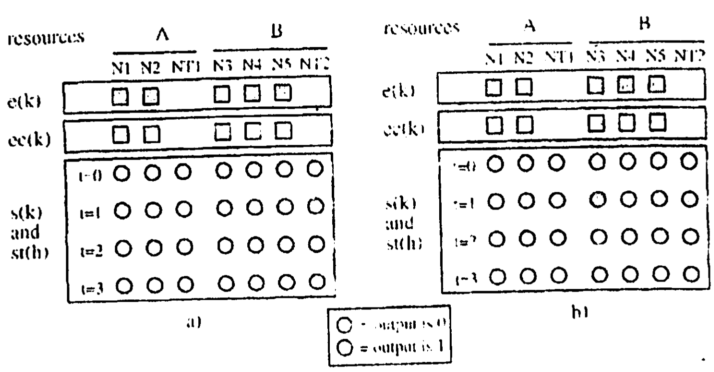


Figure17. (a) Network size (b) Inhibitory constraints

A neuron N_{ij} is assigned to each couple composed of a course c_i and of a classroom r_j. The network can be represented by a two-dimensional table.

Each line (respectively column) corresponds to a course (respectively a classroom). The neurons are located at the intersections of lines and columns (Table l(a)). The selected neurons (N_{ij}) correspond to possible choices to associate the course c_i and the classroom r_j (classrooms may be specialized or too small). In this case, their authorization input $e(i, j)$ is active. The other neurons (N_{uv}) do not play any role. The number N of neurons in the network, for a number n_c of courses and a number n_r of classrooms, is:

$$N = n_c \cdot n_r \tag{17}$$

The inhibitory imperatives just work between chosen neurons.

The course requirements force that each chosen neuron of a line must be in shared avoidance with every one of the neurons of a similar line. Undoubtedly, a course can't occur all the while in a few study halls. Neuron N_23 is then in common avoidance with neurons N_{21}, N_{22} and N_{24}. Consequently, we name these requirements 'level'.

A similar contention is legitimate for the homeroom imperatives in light of the fact that a study hall can't at the same time get a few courses. Each chosen neuron of a section is in common prohibition with every one of the neurons of a similar segment. Neuron N_23 is then in common prohibition with neuron N_{23}. In this manner, we call these limitations 'vertical'.

These two kinds of imperatives show up normally in the neural system portrayal as a two-dimensional table.

The chose neuron N_{ij} relates to the courses c_i in the study hall r_j with the understudy bunch g_v and the instructor t_v. The limitations because of the understudy gatherings and to the instructors force that the neuron N_{ij} is in shared prohibition with all the chose neurons N_{kl} requiring either a similar understudy gathering or a similar educator. For instance, the neuron N_{23} compares to the course c_2 occurring in study hall r_3 (Table 2(a)). This course c_2 is given to the understudy bunch g_1 by the instructor t_2 (Table 2(b)). As it were, neuronN_{23} is in shared rejection with all the chose neurons related with the courses given by a similar instructor t_2 (course c_4), likewise with all the chose neurons related with courses for a similar understudy bunch g_1 (courses c_l and c_4). Neuron N_{23} is then in common prohibition with neurons N_{11}, N_{12}, N_{41} and N_{44}. On account of the two-dimensional table neural system portrayal, these requirements are said to be 'corner to corner'.

Classrooms / Courses	r_1	r_2	r_3	r_4
c_1	N_{11}	N_{12}	N_{13}	N_{14}
c_2	N_{21}	N_{22}	N_{23}	N_{24}
c_3	N_{31}	N_{32}	N_{33}	N_{34}
c_4	N_{41}	N_{42}	N_{43}	N_{44}

(a)

Groups/teachers / Courses	g_1	g_2	t_1	t_2	t_3
c_1	yes	yes	yes	—	—
c_2	yes	—	—	yes	—
c_3	—	yes	—	—	yes
c_4	yes	yes	—	yes	—

(b)

Table 2. Portion limitations model: (a) neural system for four courses and four study halls, (b) requirements because of two understudy gatherings and three educators.

Every one of the associations originating from neuron N_{23} are spoken to in Fig. 18. For the two-dimensional system that we have constructed, the potential $p(i,j)$ and the yield $s(i,j)$ of neuron N_{ij} derived from (4) and (6) are separately:

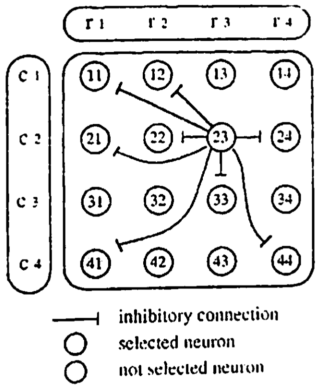
$$p(i,j) = e(i,j) - \sum_{u=1}^{+nc} \sum_{v=1}^{+nr} w(i,j,u,v) \cdot s(u,v) \tag{18}$$

and

$$s(i,j) = min[1, ec(i,j) + max[0, e(i,j) - \sum_{u=1}^{+nc} \sum_{v}^{+nr} = 1]w(i,j,u,v) \cdot s(u,v)] \tag{19}$$

Where n_c and n_r speak to separately the quantity of courses and the quantity of homerooms.

As in Section 4.2.2, it is conceivable to confine the expansion of the quantity of associations. An inhibitory neuron NH_i (resp. NV_j) is presented for each line (resp. segment): it relates to level (resp. vertical) limitations between chose neurons of a similar line (resp. segment). An extra layer of inhibitory



System case of sort 1: the inhibitory associations, just originating from neuron N_{23} for reasons of lucidity, speak to a piece of the imperatives for the case of Table 2.

neurons (ND_i) is acquainted with license the dispersion of corner to corner limitations (Fig. 19).

As beforehand, the double yield of the inhibitory neuron goes quickly to 1 since in any event one of the excitatory sources of info is equivalent to 1. The yields of neurons NH_i and NV_j concluded from (18) are:

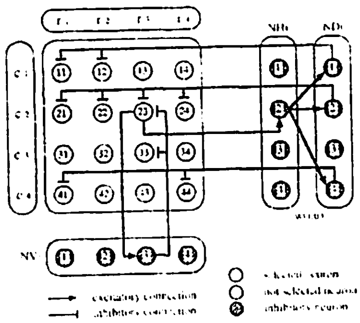
$$sh(i) = min[1, max[0, \sum_{j=1}^{+nr} s(i,j)]], \tag{20}$$

$$sv(j) = min[1, max[0, \sum_{j=1}^{+nc} s(i,j)]], \tag{21}$$

The excitatory association grid $W[w(i,u)]$, which ties the layers of neurons NH_i and ND_i, is worked with Table 2. The yield of neuron ND_i is:

$$sd(i) = min[1, max[0, \sum_{u=1}^{+nc} w(i,u) \cdot sh(u)]. \tag{22}$$

The excitatory association grid $W[w(i,u)]$, which ties the layers of neurons NH_i and ND_i, is worked with Table 2. The yield of neuron ND_i is:



$$s(i,j)=min[ec(i,j)+max][0,e(i,j)\text{-}min(sv)]$$

$$s(i,j) = min[ec(i,j) + max[0, e(i,j) - min(sv(j) + sd(i))]] \tag{23}$$

The models we have portrayed license effectively to consider the issues of inaccessibility of study halls or educators. For each availability, it is conceivable to counsel a table, called table of inaccessible assets, to restrict the utilization of the related neurons, if vital, utilizing the approval input e(i, j). The practical conduct stays indistinguishable from the one introduced in Section 4.2.1.

6. SOFTWARE FOR COMPUTER

6.1 General description

To try different things with our methodology, a product written in C language has been created. We have executed our product on a PC since it is a wide spread machine. We will see later that the exhibition moderately mean of such a machine is adequate for our application. This product permits the TS of the three school a very long time in a building establishment, however it can likewise be adjusted for some other timetable as often as possible experienced in French colleges. As an exhibit, we have picked the accompanying assets essentially for five days: 30 courses (10 courses for each school year), 10 study halls, 3 of which are specific for functional work, 6 understudy gatherings (2 bunches for each school year) and 4 educators. Here, we utilize the word 'courses' in a general importance: it implies apathetically courses, activities or functional work. Right off the bat, we pick a straightforward guide to explain the issue, however which does not infer any confinement to the extent technique or programming are concerned. There is no confinement on timetable term (week, semester or year).

(1) Constraints

A first table named courses/homerooms imperatives table (Fig. 26), shows the course list in the main segment and the homeroom list on the primary line. This table is identical to Table 1(a). All the potential cases in partner a course c_i in the study hall r_j might be chosen.

A first table named courses/homerooms imperatives table (Fig. 26),

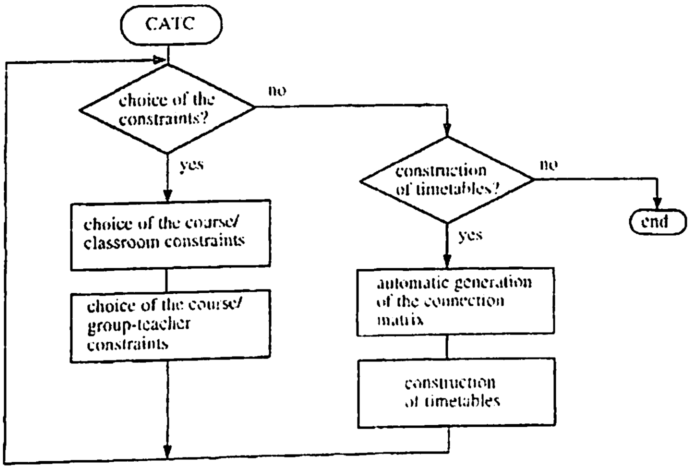


Figure 18. Software flowchart for Computer Aided Timetable Scheduling

shows the course list in the main segment and the homeroom list on the primary line. This table is identical to Table 1(a). All the potential cases in partner a course c_i in the study hall r_j might be chosen.

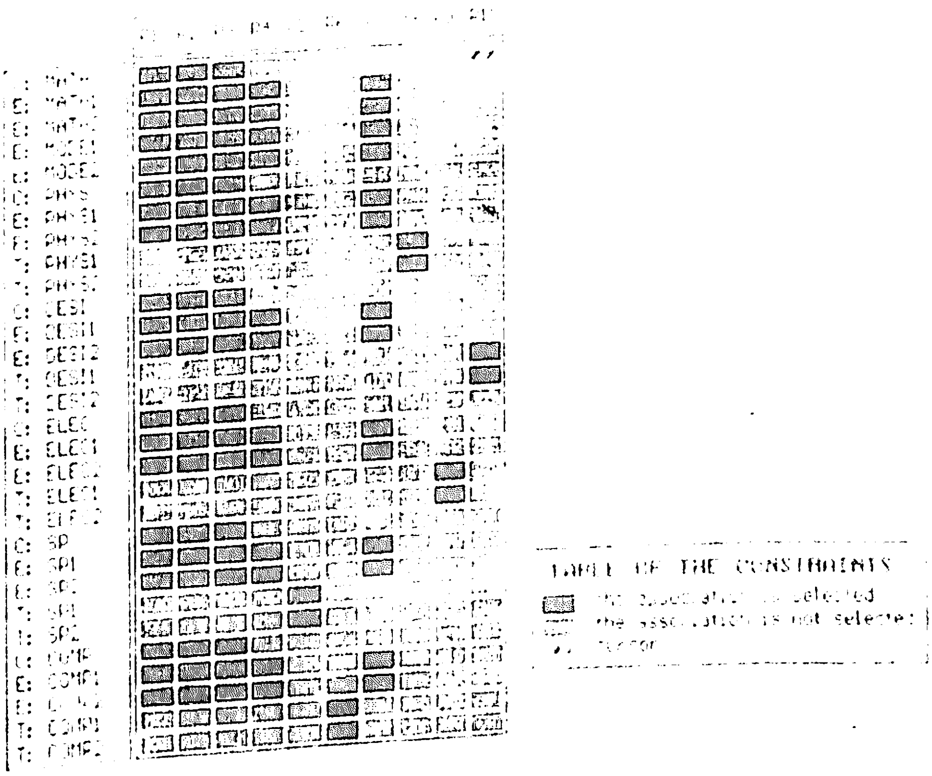


Figure 19. Table of course/classroom constrains

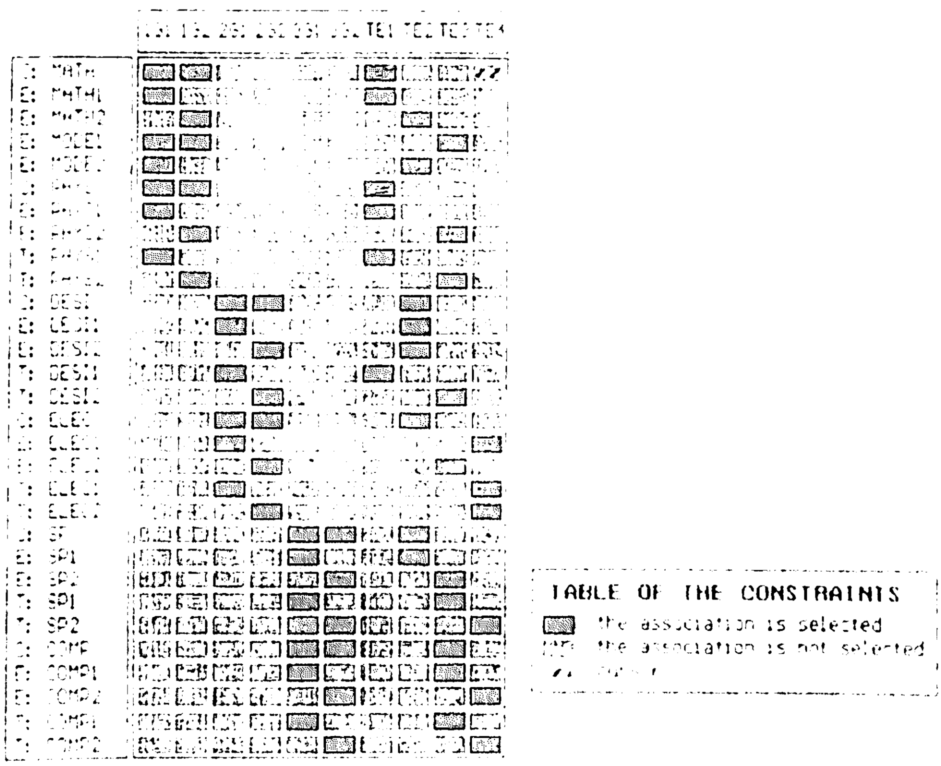


Figure 20. Table of course/classroom-teacher constrains

(2) Timetable development

As a fundamental, the product consequently assembles the system with its associations (for 'flat', 'vertical' and 'corner to corner' imperatives) with the data contained in the two past requirement tables.

At that point, the timetable being built (timetable for the main, second or third year as picked) and the table, named table of the potential decisions, show up on the screen all the while (Fig. 29). The table of the potential decisions displays the neural system yield states worked with the requirements table of the initial segment.

The TS is made as following: First, an availability (for instance, Tuesday 15 to 16 o'clock) is chosen in the timetable. At that point, the product consequently refreshes the table of the potential decisions of this vacancy. This is finished considering all the assignment requirements and all the past decisions in different timetables for a similar schedule vacancy. Thereafter, one arrangement (course c_i in the study hall r_j) is chosen among all the potential decisions given by the product. Consequently, the originator keeps control of

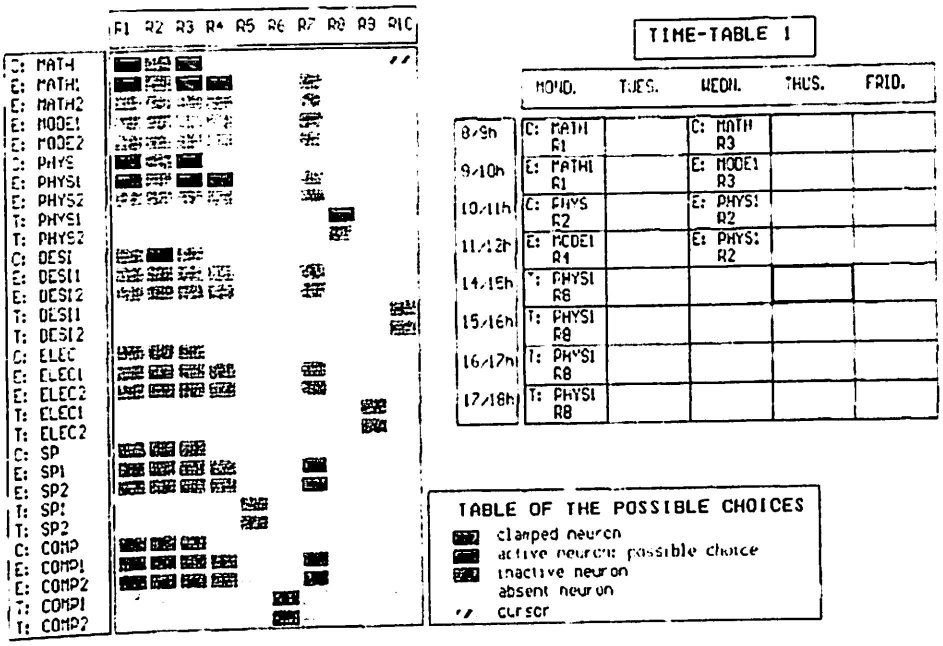


Figure 21. Table of the potential decisions and timetable during the development

the entire booking process and, as we have referenced, he is discharged from the portion limitations. This product is especially easy to understand and adaptable. In fact, it is conceivable:

- to fill, for every timetable, the vacancies in any request (for instance, to calendar long functional work toward the evening first),

- to begin the development of another timetable without having finished the past one (for instance, to convey the reasonable work for the three school a long time without focusing on the courses first),

- to include (or expel) one or a few limitations during the TS. This adjustment is then just considered for the accompanying activities. In this way, the originator can return during a TS, if something does not fulfill him. He can likewise present transitory changes on a completed timetable and test substitution arrangements, for instance, when an educator is incapacitated.

6.2 Results

Two execution criteria are fundamental for our product:

- the quantity of neurons in a system and all the more accurately the size of the memory related with the associations network. This trademark gives data about the limit of such a system to code an extraordinary number of limitations and after that to take into account the development of complex timetables,

- the calculation time for the refreshing of the neuron yields. This parameter is fundamental so as to think about ongoing applications.

6.2.1 Memory size related with the associations framework

Computation time Give us a chance to decide the quantity of fundamental associations and the size of the memory related for the systems of sort I and 2 (improved adaptation).

- Network of sort I (Figure 21)

The number N of neurons in the system, for a number n^c of courses and a number nr of study halls is given by (11). This system might be considered as totally interconnected and the number n of associations is:

$$n = N^2 = (n_c \cdot n_r)^2 \tag{24}$$

- Network of sort 2 (Figure 21) Assuming that n_c is the quantity of courses and n the quantity of study halls, we can compute the number N' of neurons by the accompanying:

- the quantity of neurons N_{ij} allocated to courses and study halls is $'n_c{\cdot}n_{r'}$,
- the quantity of inhibitory neurons for flat (NH_i) and corner to corner (ND_i) requirements is $'2 \cdot n_{c'}$,
- the quantity of inhibitory neurons for vertical limitations (NV_j) is $'n_{r'}$.

Along these lines, we can conclude the last recipe:

$$N' = n_c \cdot n_r + 2 \cdot n_c + n_r \tag{25}$$

We can likewise ascertain the number n' of associations by the accompanying:

- every neuron N_{ij} has 2 inhibitory associations: initial one with the neuron ND_i and the second one with the neuron NV_j, which suggests $2 \cdot n_c \cdot n'_r$ associations,

- every neuron N_{ij} has 2 excitatory associations: the first with the neuron NH_i and the second one with the neuron NV_j , which suggests '$2 \cdot n_c \cdot n'_r$ associations,

- the excitatory associations between each couple (NH_i, ND_u), which infers' n_c^2, associations.

Along these lines, we conclude the last recipe:

$$n' = 2 \cdot n_c \cdot n_r + 2 \cdot n_c \cdot n_r + n_c^2$$

$$n' = 4 \cdot n_c \cdot n_r + n_c^2. \tag{26}$$

Every association weight might be coded utilizing one byte (character's configuration in C language). Since the associations are paired, we can improve the coding by utilizing one piece for every one. For this situation, a significant decrease of the required memory is acquired. Table 2 gives a few estimations of the quantity of neurons and of the quantity of associations for system instances of sort 1 and 2. The quantity of associations is proportionate to the required number of memory bits. System of sort 2 (and our product) grants to oversee significant timetables (around 1700 courses and 100 study halls) with a PC, broadly accessible, having just 640 Kb or 5.2 Mb of memory (RAM). On a fundamental level, the numbers N_g of understudy gatherings and n_t of instructors are not constrained. For around 1700 courses and 100 homerooms, these numbers might be for instance equivalent to $n_g = 40$ and $n_t = 300$.

6.2.2 Computation time

The refreshing of the neuron yields, accomplished by applying connection (9), requires various counts and a length which increment with system measure $(O(n^2))$. For all intents and purposes; the refreshing strategy is firmly quickened by the decrease of the calculations since the entire count of every

Courses and classrooms		Network of type 1		Network of type 2	
n_c	n_r	N	n	N'	n'
30	10	300	90 K	370	2.1 K
300	30	9 K	81 MEG	9.63 K	126 K
1700	100	170 K	28.9 G	173.5 K	3.57 MEG

Table 3. The number (N,N') of neurons and the number (n,n') of connections for network example of type 1 and type 2

capability of neurons ND_i, NH_i and NV_j is commonly pointless. To be sure, this count can be halted when the outcome achieves esteem 1 in light of the fact that the neuron yield can't surpass this worth. In this manner, the product continues to the system yield refreshing in an extremely brief time (short of what one moment for 800 courses, 50 study halls, 20 understudy gatherings and 150 instructors), which establishes a semi ongoing apparatus at the fashioner's time scale.

6.2.3 Comparisons

An unpleasant examination can be made between our outcomes and the ones gotten by Tabu hunt system [9] and Potts neural systems [7]. The strategies, referenced previously, propose likewise an answer for TS issues for their own college framework (resp. in Switzerland and Sweden) as indicated by their particularities. They have some regular focuses between them, basically, coding the biggest conceivable number of requirements and advancing a cost capacity (Section 3), with no human communication. So as to acquire timetables in a sensible brief time (a couple of moments) the utilization of a centralized computer PC is compulsory. For example, Tabu inquiry procedure needs 11 seconds on CDC Cyber 170/855 for 288 courses, 67 study halls, 1729 understudies and 143 instructors, yet Potts neural system requires around one moment on a CRAY XMP for 50 week after week hours, 60 study halls, 45 "classes" or understudy gatherings, and 90 educators. Despite what might be expected, our framework is intelligent and utilizes a PC. The hard limitation the executives is acknowledged in an incredibly performing way. It establishes an amazing asset for extraordinary timetables (for instance, 1700 courses, 100 study halls, 40 understudy gatherings and 300 educators), running on a generally spread PC. We should see additionally that for the two previously mentioned procedures, the presentation of the delicate limitations before the enhancement procedure establishes a long assignment, which isn't considered in the calculation time. In our methodology, the human master does not need to present delicate imperatives, yet he verifiably considers during the intuitive timetable development.

7. Conclusion

In this work, our way to deal with undertaking planning with disjunctive imperatives utilizing the inside challenge properties of a neural system with Lateral Inhibitory Networks (LIN) has been depicted. It has been connected to the perplexing instance of Timetable Scheduling (TS) and all the more exactly to class TS. We have demonstrated that the methodology permits TS with an incredible number of imperatives utilizing a straightforward PC. Our product permits the elaboration of complex school timetables.

Some corresponding apparatuses can be acquainted all together with assistance the fashioner utilizing the school TS. One of them is the programmed topping off of the imperative tables attributable to, for instance, the sign of the study hall limits. Another is programmed the board obviously recurrence and the treatment of special cases because of tests and occasions. For this situation, the product CS could be combined with a spreadsheet. The PC offices can be utilized to print the timetables for understudies as well as for study halls and educators.

The general methodology that we have depicted, permits to consider various expansions when disjunctive limitations happen. Some of them are the development of various sorts of non school timetables, and the undertaking planning for different facilitates (for instance, in workshops).

References

[1] Holland J. H. Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence.— London: Bradford book edition. 1994 — 211 p.

[2] De Jong K. A. Genetic Algorithms: A 10 Year Perspective //In: Procs of the First Int. Conf. on Genetic Algorithms. 1985. —pp.167—177.

[3] О некоторых теоретических аспектах экологического мониторинга состояния природной среды / Ю. А. Израэль и др. // В кн.: Проблемы экологического мониторинга и моделирование экосистем. Том II. – Л.: Гидрометеоиздат. 1979. — сс. 7-29.

[4] Автоматизация поискового контрулирования (искусственный интеллект в машинном проектировании) / Под ред. А. И. Половинкина. — М.: Радио и связь. 1981. — 344 с.

[5] Семевский Ф. Н., Семенов С. М. Математическое моделирование экологических процессов. — Л.: Гидрометеоиздат. 1982. — 326 с.

[6] Aleksander I., Morton H. An Introduction to Neural Computing. — London: Chapman and Hall. 1990.

[7] McCulloch W. S., Pitts W. A logical calculus of ideas imminent in nervous activity // Bulletin Mathematical Biophysics. — 1943. — 5. — pp.115—133 (Имеется перевод: Дж. Маккаллок, У. Питтс. Логическое исчисление идей, относящихся к нервной деятельности // В кн.: Автоматы. – М.: ИЛ. 1956)

[8] Hebb D.O. The Organization of Behavior: A Neuropsychological Theory. — NewYork: Wiley. 1949.

[9] Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain // Psychological Review. — 1958.— 65.— pp.386—407 (Розенблатт Ф. Принципы нейродинамики. — М.: Мир, 1966. — 480 с.).

[10] Minsky M. L., Papert S. A. Perceptrons. Cambridge, MA: MIT Press, 1969 (Имеется перевод: Минский М., Пейперт. Персептроны. – М.: Мир, 1971. — 261с.).

[11] Rumelhart D. E., Hinton G. E., Williams R. J. Learning internal representation by error propagation // In: D.E.Rumelhart and J.L.McClelland

(Eds.) Parallel Distributed Processing. Vol. I Foundations. – Cambridge, MA: MIT Press. 1986. pp.318–362.