Minimal BSDT Abstract Selectional Machines and Their Selectional and Computational Performance

Petro Gopych
Universal Power Systems USA-Ukraine LLC, 3 Kotsarskaya st., Kharkiv 61012, Ukraine
pmg@kharkov.com

Abstract. Turing machine (TM) theory constitutes the theoretical basis for contemporary digital (von Neumann) computers. But it is problematic whether it could be an adequate theory of brain functions (computations) because, as it is widely accepted, the brain is a selectional device with blurred bounds between the areas responsible for data processing, control, and behavior. In this paper, by analogy with TMs, the optimal decoding algorithm of recent binary signal detection theory (BSDT) is presented in the form of a minimal one-dimensional abstract selectional machine (ASM). The ASM's hypercomplexity is explicitly hypothesized, its optimal selectional and super-Turing computational performance is discussed. BSDT ASMs can contribute to a mathematically strict and biologically plausible theory of functional properties of the brain, mind/brain relations and super-Turing machines mimicking partially some cognitive abilities in animals and humans.

Keywords: neural networks, Turing machine, brain, memory, consciousness

1 Introduction

It is widely accepted that no possible abstract [1] or physically realizing [2] computational device can be more powerful than Turing machine (TM) and this fact is actually the formal theoretical substantiation for the construction of contemporary digital (von Neumann) computers. But lately it has been speculated that in nature such physical processes may exist that TM computational abilities are insufficient for their simulations. The most famous hypothesis states that Turing computations are not strong enough to model human intelligence though there is no consensus of opinions on what human mind super-Turing abilities could actually mean [3,4,5].

On the other hand, it is widely accepted that the brain is a *selectional* device [6,7] which, in contrast to von Neumann computers, has blurred bounds between its hierarchically constructed and often overlapped functional areas and runs as a learnable system continuously adapting to its environment, not under the control of a given program. The brain demonstrates also its high tolerance to errors (noise) in data and damages to computational devices and can provide multiple correct solutions to a given problem. Additionally, the brain is profoundly influenced by the human genome that defines, to great extent, human dynamic behavior and cognitive abilities [7,8]. The traditional concept of information does not consider the meaning of information but in biology it may be clearly defined (that is what 'was selected') though, due to the multilevel brain hierarchy, this meaning is extremely difficult to specify [7].

In recent years, analog recurrent neural networks (ARNNs) were intensively studied as possible super-Turing abstract computational devices and it was demonstrated that with real-valued weights they can outperform standard TM in terms of its computational power [9]. ARNNs are in part motivated by the hypothesis that real neurons can hold continuous values with unbounded precision because of continuity of their underlying physical and chemical processes. Now the idea of analog brain computations continues to attract new adherents [10], though most experts doubt whether it is possible to implement analog devices with weights of unbounded precision (e.g., because of unavoidable noise, finite dynamic range, energy dissipation, or as 'the strength of the synapse is not very important, once it is large enough' [11]).

Taking into account that the brain is a selectional device, by analogy with TMs [1], we propose another approach in this paper: a brain-specific biologically relevant computational formalism of *abstract selectional machines* (ASMs), devices extracting from their input a given message (pattern or image). For this purpose, we transform optimal decoding algorithms of recent binary signal detection theory (BSDT) [12-17] into ASMs of different types and find their *optimal* selectional as well as *super-Turing* computational performance. The BSDT ASM's *hypercomplexity* is explicitly hypothesized; biological plausibility of ASMs, their parallels in TMs and some other abstract devices (e.g., [9, 18]) are discussed.

2 BSDT Coding/Decoding and Performance

The BSDT [12-17] operates with N-dimensional vectors x with their components $x^i = \pm1$, a reference vector $x = x_0$ representing the information stored or that should be stored in a neural network (NN), binary noise $x = x_r$ (the signs of its components are randomly chosen with uniform probability, ½), and vectors $x(d)$ with components

$$x_i(d) = \begin{cases} x_0^i, & if \quad u_i = 0, \\ x_r^i, & if \quad u_i = 1 \end{cases} \qquad d = \sum u_i / N, \quad i = 1,..., N \tag{1}$$

where u_i is 0 or 1. If m is the number of marks $u_i = 1$ then $d = m/N$, $0 \le d \le 1$; d is a fraction of noise components in $x(d)$, $q = 1 - d$ is a fraction of intact components of x_0 in $x(d)$ or an *intensity of the cue*, $0 \le q \le 1$. If $d = m/N$, the number of different $x(d)$ is $2^m C^N_m$, $C^N_m = N!/(N - m)!/m!$; if $0 \le d \le 1$, this number is $\sum 2^m C^N_m = 3^N$ ($m = 0, 1, \ldots, N$). As the set of $x(d)$ is complete, always $x = x(d)$.

The data coded as described are decoded by a two-layer NN with N model neurons in its entrance and exit layers which are linked by the rule 'all-entrance-layer-neurons-to-all-exit-layer-neurons.' Its synapse matrix elements are $w_{ij} = \xi x^i_0 x^j_0$ where $\xi > 0$ ($\xi = 1$ below), $w_{ij} = \pm1$. That is a perfectly learned intact NN storing *one* reference pattern x_0 only. The NN's input $x = x_{\text{in}}$ is decoded (x_0 is identified in x_{in}) successfully if x_{in} is transformed into the NN's output $x_{\text{out}} = x_0$ (such an x_{in} is called a *successful input*, x_{succ}); an additional 'grandmother' neuron (an integrate-and-fire coincidence neuron responding to a precise combination of its inputs, x_0) checks this fact. The weighted sum of all inputs to the jth exit-layer neuron is $h_j = \sum w_{ij} x^i_{\text{in}}$ where x^i_{in} is an input/output signal of the ith entrance-layer neuron, a fan-out that conveys its input to all exit-layer neurons. The output of the jth exit-layer neuron is

$$x_{out}^{j} = \begin{cases} +1, & if \quad h_j > \theta \\ -1, & if \quad h_j \le \theta \end{cases} \qquad (2)$$

where $\theta \ge 0$ is the neuron's triggering threshold (for $\theta < 0$ see ref. 14), the value $x^{j}_{\text{out}} = -1$ at $h_j = \theta$ was arbitrary assigned. If $x^{j}_{\text{out}} = x^{j}_{0}$ (j = 1, …, N) then x_{in} is x_0 damaged by noise; otherwise, it is a sample of noise, x_{r}. The above NN decoding algorithm can also be presented in functionally equivalent convolutional and Hamming distance forms each of which is the best in the sense of pattern recognition quality [12,14,17].

For intact perfectly learned NNs, decoding probability of vectors $x = x(d)$, $d = m/N$, can be calculated analytically [12,17]:

$$P(N,m,\Theta) = \sum\nolimits_{k=0}^{K} C_k^m / 2^m, \; K_0 = \begin{cases} (N-\Theta-1)/2, & if \quad N \quad is \quad odd \\ (N-\Theta)/2-1, & if \quad N \quad is \quad even \end{cases} \qquad (3)$$

where Θ is an even integer θ, $-N \le \Theta < N$; if $K < K_0$ then $K = m$ else $K = K_0$ (in this context k is the Hamming distance between x and x_0, K is its threshold value, and K_0 is the K for a given Θ). If $\Theta < -N$ then $P(N,m,\Theta) = 1$, if $\Theta \ge N$ then $P(N,m,\Theta) = 0$. For any $\theta \in \Delta\theta_j$, the NN decoding algorithm (see Eq. 2) gives $P(N,m,\theta) = P(N,m,\Theta_j)$ where $\Theta_j \in \Delta\theta_j$ (here, $j = 0, 1, 2,\dots, N+1$, $\Theta_j = 2j - N - 1$). If $0 < j < N + 1$ then $\Delta\theta_j = [\Theta_j - 1, \Theta_j + 1)$ and $\Delta\theta_j = [\Theta_j, \Theta_j + 2)$ for odd and even N, respectively; if $j = 0$ and $j = N + 1$ then $\Delta\theta_0 = (-\infty,-N)$, $P(N,m,\Theta_0) = 1$ and $\Delta\theta_{N+1} = [N,+\infty)$, $P(N,m,\Theta_{N+1}) = 0$. As BSDT decoding algorithm exists in three equivalent forms, many of its parameters can be calculated one through the other for a given N, e.g.: m, d, and q or Q (a convolution of x and x_0, $Q = \sum x^i x^i_0$, $-N \le Q \le N$), ρ (correlation coefficient, $\rho = Q/N$), k (Hamming distance, $k = (N + Q)/2$), Θ, Θ_j, $\theta \in \Delta\theta_j$, F_j (false-alarm probability or the probability of identification of a noise vector $x = x_{\text{r}}$ as x_0), and j (confidence level of decisions) [13,16]. For this reason, decoding probability (Eq. 3) can be written in some equivalent forms (as functions of different sets of their parameters): $P(N,m,\Theta) = P(N,m,\theta) = P(N,m,\Theta_j) = P(N,d,j) = P(N,d,F) = P(N,q,j) = P(N,q,F)$ etc.

3 Minimal BSDT ASMs and Their Functions

The BSDT explicitly defines a finite set of optimally coded objects, $x = x(d)$, and a finite-sized tool for their optimal decoding, a learned NN. Consequently, it is suitable for solving two broad classes of practical problems: studying a set of x given the NN (data mining, e.g. [19]) and studying an NN with the set of x given (memory modeling, e.g. [15,17]). To match these problems better, BSDT optimal decoding algorithm can be presented in either a feedforward form (the case of data mining) or a cyclic form (the case of memory modeling). The former and the latter correspond to *passive* ASMs (Section 3.1) and *active* ASMs (Section 3.2). For simplicity, *one-dimensional* (dealing with one-dimensional inputs) ASMs will be only considered.

3.1 Minimal One-dimensional BSDT Passive ASM

First, by analogy with TMs, we present BSDT decoding algorithm studying a set of vectors x given the NN, as a minimal one-dimensional BSDT *passive ASM* (PASM, Fig. 1A; cf. Fig. 2 of ref. 15). It consists of an N-channel scanner (box 1), the learned NN (box 2), and grandmother neuron (diamond 3). The PASM is also supplied by a *finite-length* one-dimensional *read-only* data tape divided into equal cells bearing binary signals only, +1 or −1. The tape is *movable* (with its drive outside the PASM) while the scanner stays still, that is '*passive*.'

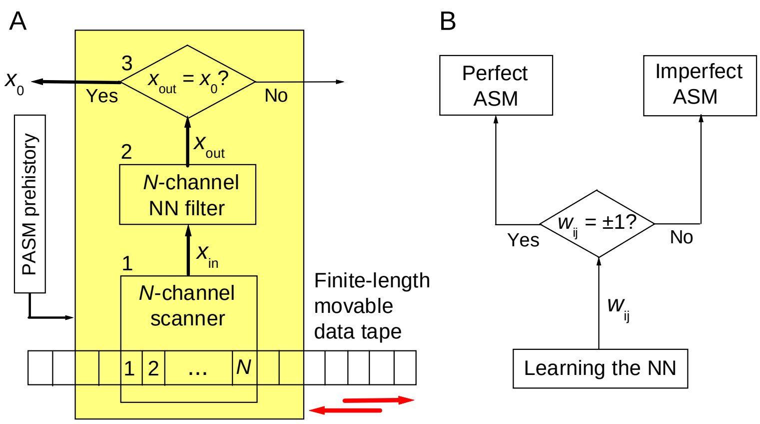

Fig. 1. Minimal one-dimensional BSDT PASM and conditions defining its selectional (classification) quality. **A**, The architecture of PASMs; arrows near the tape, possible directions of its movement; thick arrows, pathways for transferring vectors x (groups of N synchronous spikes); learned NN (box 2) plays the role of a filter tuned to select x_0; given its prehistory, each individual PASM can run in isolation. **B**, Influence of NN synaptic weights on ASM selectional performance ($w_{ij} = \pm1$ and $w_{ij} = \pm\xi$, if $\xi > 0$, are functionally equivalent).

The scanner (box 1) reads out simultaneously N successive signs ±1 constituting together the PASM's input string x_{in}, transforms x_{in} into its active form (a set of N synchronous pulses or 'spikes' [17]), and conveys it to the learned NN (box 2). Here, x_{in} generates the NN's output x_{out} and, then, grandmother neuron (diamond 3) checks whether $x_{out} = x_0$. If that is the case, then $x_{in} = x_{succ}$, and x_0 becomes available for its further use in its active form. Afterwards, the tape shifts left or right[1], the scanner reads out next x_{in} and so forth, generally in a never-stop regime defined by the outside driver. *Feedforward* processing of each input x_{in} always gives the definite PASM selectional decision (that is the PASM's goal) during a finite time period, Δt_{PASM}.

All the PASM's internal connections as well as its environmental disposition were hardwired and all its parameters (N, x_0, w_{ij}, θ) were adjusted during its design process. This process (and its final product, a PASM) is defined by the PASM *prehistory*

[1] To write out 2^N different N-dimensional vectors x_{in} (strings of the size N of 1s and −1s), it is enough to have in general the tape of $2^N + N - 1$ cells. If so then these 2^N strings have to be read out in a definite manner only: for example, from left to right by shifting the tape in the scanner one cell left after each act of reading. If the tape of $2^N N$ cells is used then any x_{in} may be read out after the reading any other x_{in}.

which, as we suppose, may be loosely divided into its evolutionary (genome-specific) and developmental (experience- or learning-specific) stages. If the environment changes then the PASM may be adapted (by outside factors) to new conditions but resulting PASM becomes already another one having already another prehistory.

A perfectly learned (with $w_{ij} = \pm 1$) BSDT NN produces selections the best in the sense of pattern recognition quality (Section 2). Hence, in contrast to ARNNs [9], for the construction of PASMs, NNs with rational or real weights are not required (in this case they lead to more complicate computations only). Of computational viewpoint a binary NN (as box 2 in Fig. 1A) is equivalent to a *finite automaton*, the simplest TM [9]. For verifying this NN's outputs the PASM uses the string x_0, a PASM specific *advice* common for all the inputs x_{in}. Consequently, the PASM may be considered as an advice TM [9] with advice sequence x_0 completely defined by PASM prehistory (the length of x_0, N, is the advice TM's noncomputability level [9]). If x_0 is known then any PASM, using it, may be simulated by a TM which, because any PASM is specified by its prehistory at a process level beforehand (as x_0 and a given pattern of NN connections), will spend for simulating the PASM selections the time $\Delta t_{\text{TM}} > \Delta t_{\text{PASM}}$. In that sense PASMs are *super-Turing* computational devices.

A contradiction arises: on the one hand, advice TMs define a nonuniform class of noncomputability [9] while, on the other hand, the same TMs substantiate the PASMs having super-Turing computational abilities. To resolve this contradiction, we recall that the tape does not contain any PASM instructions and the PASM's architecture, parameters, and advice were completely hardwired ('programmed') in the course of PASM prehistory, *before* the moment when the PASM was placed into operation (it is possible because all PASM inputs are finite-sized and their total amount is limited). Thus, the source of PASM hypercomputability is *decoupling* between programming and computations: all PASM computations are completely specified ('programmed') during the PASM prehistory which, in contrast to TM specifications, has uncertain length and may be in general of infinite length (not describable by finite means).

3.2 Minimal One-dimensional BSDT Active ASM

Here, by analogy with TMs we present BSDT decoding algorithm studying an NN with the set of its inputs x given, as a minimal one-dimensional BSDT *active ASM* (AASM, Fig. 2; cf. Fig. 2 of ref. 15). It consists, in particular, of an N-channel movable (that is *'active'*) scanner (box 1), the learned NN (box 2), and grandmother neuron (diamond 3). The AASM is also supplied by an *infinite* one-dimensional still *read-only* data tape divided into equal cells bearing binary signals only, +1 or −1. This part of the AASM is actually a PASM (Fig. 1A) though AASMs additionally have the scanner's drive (box 6, see also footnote 1). Boxes 5 and 8 count NN inputs, diamonds 3 and 4 are the points of choice. It is supposed that internal loop (1-2-3-4-5-6-1) runs due to the scanner's *regular shifts* while external loop activation (1-2-3-4-7-8-6-1) leads to *a skip* of the scanner (in such a way any distant region on the tape becomes almost immediately available for the search; whether or not the skip is needed, its size and direction are defined by an *ASM society*, box 9).

To identify among AASM inputs x_{in} the one that is x_{succ}, the AASM successively examines each next x_{in} produced by one-step shift of the scanner. Once x_{succ} happens (i.e., $x_{out} = x_0$ in diamond 3), the search is finished, x_0 in its active form [17] is passed to ASM society (box 9), and the AASM is ready for searching for the next x_{succ}. If among i inputs produced by i successive regular scanner shifts there is no x_{succ} and i exceeds its upper bound[2], i_{max}, then diamond 4 interrupts internal loop and requests for *an advice* whether or not to continue the search (diamond 7). If the advice produced by ASM society (box 9, a counterpart to an 'oracle' [9] of TM theory) is 'to continue' then the scanner skips and the loop 1-2-3-4-5-6-1 is again activated.

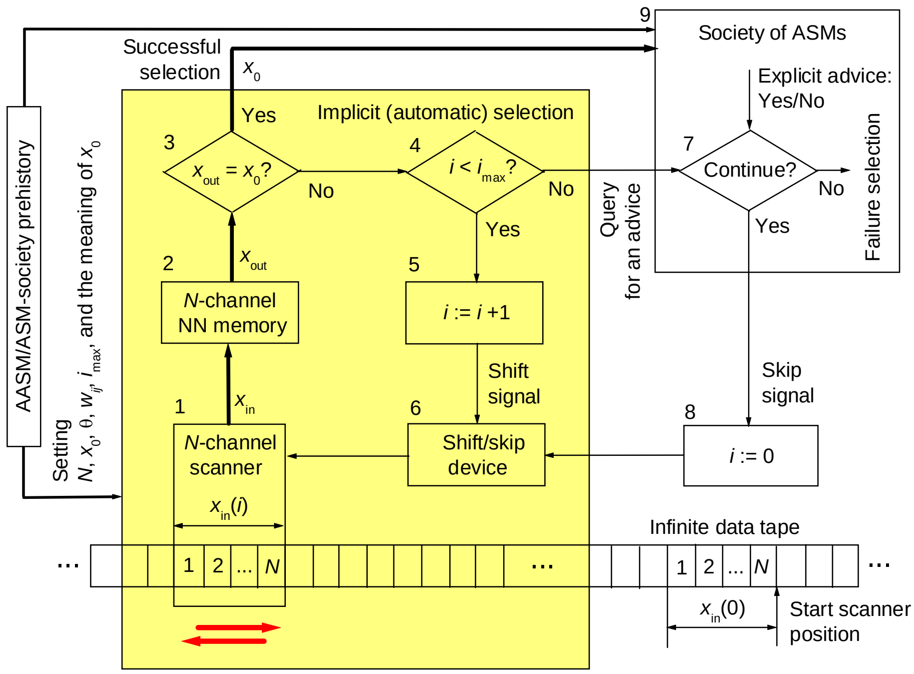

Fig. 2. The architecture of minimal one-dimensional BSDT AASMs. Arrows near the scanner, its possible movement directions; thick and thin arrows, pathways for vectors x and connections for transferring control asynchronous signals, respectively; learned NN (box 2) plays the role of memory unit storing the x_0; areas $x_{in}(0)$ and $x_{in}(i)$ on the tape are the initial and the ith input string, respectively; an individual AASM can run if it is only a member of the ASM society (a so far unspecified set of interactive ASMs equipped by sensory and executive devices, box 9).

Everything what concerns PASM selectional and computational performance (Section 3.1) is valid for the part of an AASM shaded in Fig. 2. But in spite of the AASM's optimality, no x_{succ} may be discovered, simply because the series of inputs consists of failure strings only (of x_{in} generating $x_{out} \neq x_0$). Hence, the selectional process (by shifting the scanner) may never stop in general and the AASM's goal (discovering the x_0) may not be achieved in finite time — that is the *AASM halting problem*. To solve it, the search strategy should be sometimes changed according to an external advice. For AASMs, such an advice is produced by ASM society having a

---

[2] Parameter i_{max} is specified by AASM prehistory. For possible neurobiological reasons to fix i_{max}, see ref. 12, 15, 17; the BSDT natural value of i_{max} is the amount of different $x(d)$, 3^N.

hyperselectional power, i.e., having the capability to generate the advice, in spite of its TM noncomputability, during a finite time period, Δt_{adv}. In reply to the query of diamond 4, in the milieu of ASM society, thanks to its collective properties and hypercomputational power of PASMs in AASM bodies, the advice is being selected (the halting problem is being solved) taking into account that it depends, we suppose, on the society's prehistory, its current assessment of rewards (successful selection) and punishments (failure selection), previous searching *history*[3], and time constraints.

The advice is the society history's 'event' produced by the society's current inputs and its current internal state given prehistory. On the other hand, this advice is also a part of particular AASM (pre)history and, consequently, the (pre)history of the whole society is simultaneously a part of (pre)history of its individual member, the AASM. Of this fact, a *hypercomplexity* of ASM/ASM-society (pre)histories follows: they are related, may span *infinitely* back in time and encompass events up to the origin of life (or even the Universe). Thus, the ASM society's hyperselectivity is ensured by its hypercomplexity: in the realm of ASMs their is no problem of hypercomputations but, instead, the problem of ASM/ASM-society hypercomplexity occurs.

4 Optimal Selectional Performance of Minimal BSDT ASMs

A PASM generates its definite selectional decision for each input, x_{in}. An AASM makes its selectional decisions when among its inputs a successful one, $x_{in} = x_{succ}$, is encountered. If PASMs and AASMs operate over the same set of their inputs then they have common selectional performance. For distinctness, below we define this performance using the set of $x_{in} = x(d)$ given a specific d or $0 \leq d \leq 1$ (Section2).

We introduce an ASM's *absolute* selectional power (SP) and *relative* SP: $\alpha(\theta,m/N)$ and $\gamma(\theta/N)$, respectively. $\alpha(\theta,m/N)$ is the amount of x_{succ} given θ and $d = m/N$: $\alpha(\theta,m/N) = P(N,m,\theta)2^m C^N_m$ where $P(N,m,\theta)$ is defined by Eq. 3 (see Section 2, Fig. 3A and its legend). $\gamma(\theta/N)$ is a fraction of x_{succ} given θ, N, and all d from the range $0 \leq d \leq 1$: $\gamma(\theta/N) = \beta(\theta/N)/\beta_{max}(N)$ where $\beta(\theta/N) = \sum\alpha(\theta,m/N)$, $\beta_{max}(N) = \sum 2^m C^N_m = 3^N$ and $m = 0, 1, 2, \ldots, N$ (Fig. 3B).

Fig. 3A demonstrates, in particular, that the largest number of x_{succ} is among inputs with an 'intermediate' m_0/N fraction of noise, though in this case the probability $P(N,m,\theta)$ of discovering the x_{succ} is not always maximal (e.g., in Fig. 3A, $m_0/N = 3/5 = 0.6$ while $P(5,3,2) = 1/2 < 1$). As Fig. 3B shows, larger $\gamma(\theta/N)$ and negative θ correspond to 'low-confidence' selections ($F > 1/2$) while smaller $\gamma(\theta/N)$ and positive θ correspond to 'high-confidence' selections ($F < 1/2$). This result is consistent with the fact that neuron thresholds $\theta \geq 0$ are preferable in practice. The region $-\infty < \theta/N < +\infty$ may also be divided into areas where the more the N the larger the $\gamma(\theta/N)$ is $[\theta/N < (\theta/N)_0]$ and the more the N the smaller the $\gamma(\theta/N)$ is $[\theta/N > (\theta/N)_0]$. Consequently, $(\theta/N)_0 \sim 0.4$ may be accepted as another definition of the border between 'low-

[3] We distinct prehistory and history. Prehistory is the process (and, eventually, the product) of designing the ASM and consists of its evolutionary (genome specific) and developmental (learning specific) stages (see also ref. 20). History is a series of ASM 'events' (outputs initiated by particular inputs) given its prehistory. Any PASM history is predictive because, given prehistory, it is completely defined by PASM inputs (Fig. 1A) while any AASM history is unpredictive and may infinitely rich become because, given prehistory, it depends on AASM inputs as well as the current state of ASM society (Fig. 2).

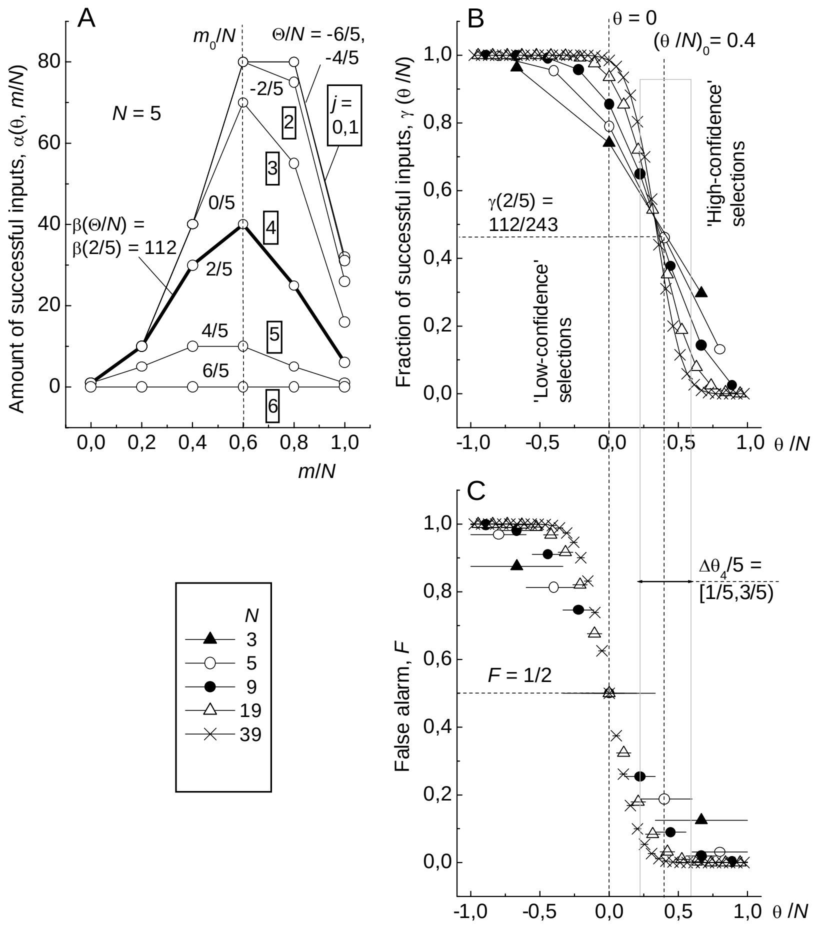

Fig. 3. Absolute (α) and relative (γ) selectional power for minimal one-dimensional BSDT ASMs. **A**, $\alpha(\theta,m/N)$, open circles connected by line segments (θ, neuron triggering threshold; m, the number of an input vector's noise components; $N = 5$); boxed numbers are decision confidence levels (the curves' serial numbers), j [16]; unboxed numbers designate middle points, Θ_j/N, of $\Delta\theta_j/N$ intervals in **C**; values of $\alpha(2,m/5)$ lie on the thick solid line (for them $\beta(\Theta_j/N) = \beta(2/5) = \sum\alpha(2,m/5) = 112$); dashed line depicts m_0/N, a fraction of input noise components at maximums of $\alpha(\theta,m/N)$. **B**, $\gamma(\theta/N) = \gamma(\Theta_j/N)$, different signs ($N = 3$, 5, 9, 19, and 39; Θ_j/N, abscissa of each sign); horizontal dashed line points to that open circle whose $\gamma(\theta/N)$ corresponds to thick line in **A**, $\gamma(2/5) = \beta(2/5)/\beta_{\max}(5)$, $\beta_{\max}(5) = 3^5 = 243$; $\gamma(\theta/N < 1) = 1$ and $\gamma(\theta/N \geq 1) = 0$ are not shown. **C**, False alarms for relative neuron triggering threshold intervals, $\Delta\theta_j/N = [(\Theta_j - 1)/N, (\Theta_j + 1)/N)$; because of their infiniteness, $\Delta\theta_{N+1}/N$ and $\Delta\theta_0/N$ are not shown; vertical dashed lines indicate $\theta = \Theta_{(N+1)/2} = 0$ and the middle point, $(\theta/N)_0$, of the boxed interval, $\Delta\theta_4/5$, that can separate 'low-confidence' and 'high-confidence' selections. In all panels signs give the function's values for all $\theta/N \in \Delta\theta_j/N$, not for separate Θ_j/N only (cf. [13,14,16]).

confidence' and 'high-confidence' selections though this border has an N-dependent non-zero width, $\Delta\theta/N$ (see Fig. 3B and C).

5 Discussion and Conclusion

The BSDT ASM's input (a string of the size N, x_{in}) is an arbitrarily chosen finite fraction of in general infinite input data set, the sought-for output (x_0) is alsow fixed. Hence, any other NN with such properties may be considered as a kind of ASM, though no one (except BSDT ASMs) can be 'ideal' [14]. In particular, that concerns ARNNs [9] which are hypercomputational machines if real-valued weights are being used. In contrast, ASMs use integer weights (Fig. 1B) and, in spite of that, provide *both* optimal selectional performance (due to BSDT optimality) and super-Turing computational ability (due to decoupling between programming and computations).

ASM x_0-specificity and the notion of genome/learning-specific ASM prehistory allow to define *the meaning* of a message 'that was selected': the meaning of a successful input, x_{succ}, is the meaning of x_0 or the content of prehistory of the ASM selected x_0, $M(x_0)$. The total amount of x_{succ} may be large (Fig. 3A, B) but, for any x_{succ}, $M(x_{succ}) = M(x_0)$. This may explain why in biology *correct decisions are multiple and degenerate* (many different inputs, x_{succ}, produce the same output, x_0). $M(x_0)$ may exist in an implicit, $M_{impl}(x_0)$, or an explicit, $M_{expl}(x_0)$, form. $M_{impl}(x_0)$ is implemented as an ASM's hardwiring and parameter setting (see also [18, 20]), that is a '*subjective'* property not available out of the system. To be communicated to other *similar* (having partially the same prehistory) system, $M(x_0)$ *or its part* should be coded/described using a (natural) *language*—a code interpreted in computation theory, e.g. [9], as a discrete function defining word/meaning relations. The code/language description obtained is $M_{expl}(x_0)$. The more elaborate the language the more complete $M_{expl}(x_0)$ may be, $M_{expl}(x_0) \to M_{impl}(x_0)$ but never $M_{expl}(x_0) = M_{impl}(x_0)$; when a finite message is being coded/decoded, *the similarity* of communicating systems (of their prehistories) ensures the common *context* for them—an implicitly available and infinite in general additional information needed for unambiguous understanding the message.

For biological plausibility of AASMs considered as memory units, see ref. 15, 17. According to them, an AASM's internal loop is interpreted as a minimal structure representing *implicit (unconscious) memory* unit (it is shaded in Fig. 2) while together with the external loop it already represents an 'atom' of *explicit (conscious) memory* or an 'atom' of *consciousness* relying on an explicit ('conscious') advice. Biological plausibility of PASMs may be illustrated by BSDT theory for vision where 'a hierarchy of tuned local NN units' (i.e., PASMs) extracts 'step-by-step from an initial image its more and more general features/properties' [15, p. 149]. Also we draw attention to a clear analogy between shift and skip movements of an AASM's scanner and, respectively, slow drifts and saccades of the human eye [21].

In sum, an original 'selectional' approach to biologically plausible network computations has been introduced. BSDT ASMs were defined and it was demonstrated that they are hypercomplex hypercomputational learnable (not programmable) NN devices providing optimal selectional performance with preferably integer synaptic weights. This approach is a natural tool for the description of brain/mind functions (spike computations) and could contribute to the construction of mathematically strict and biologically relevant future theory of brain cognitive abilities and brain/mind relations (for the first explicit example, see ref. 16). Of the technical perspective, BSDT ASMs might substantiate a large and diverse family of original high-performance artifacts (complex super-Turing selectional machines constructed of minimal learned BSDT ASMs as their building blocks, sensory input devices and executive output devices) mimicking in part different perceptual and cognitive functions in animals and humans.

References

1. Turing, A.M.: On Computable Numbers with an Application to the Entscheidungs-problem. Proceed. London Math. Soc. **42** (1936-1937) 230-265
2. Deutsch, D.: Quantum theory, the Church-Turing Principle and the Universal Quantum Computer. Proceed. Roy. Soc. London **A400** (2005) 96-117
3. Searl, J.R.: Minds, Brains, and Programs. Behav. Brain Sci. **3** (1980) 417-457
4. Penrose, R.: Shadows of Mind. Oxford University Press, Oxford, England (1994)
5. Copeland, B.B.J.: Beyond the Universal Turing Machine. Australasian J. Philosophy **77** (1999) 46-66
6. Edelman, G.M.: Naturalizing Consciousness: A Theoretical Framework. Proceed. Natl. Acad. Sci. USA **100** (2003) 5520-5524
7. Brent, R., Bruck, J.: Can Computers Help to Explain Biology? Nature **440** (2006) 416-417
8. Butcher, L.M., Kennedy, J.K.J., Plomin, R.: Generalist Genes and Cognitive Neuroscience. Curr. Opin. Neurobiol. **16** (2006) 145-151
9. Siegelmannn, H.T.: Neural Networks and Analog Computation: Beyond the Turing Limit. Birkhauser, Boston-Basel-Berlin (1999)
10. O'Reilly, R.C.: Biologically Based Computational Models of High-Level Cognition. Science **314** (2006) 91-94
11. Mander, E., Goaillard, J.-M.: Variability, Compensation, and Homeostasis in Neuron and Network Function. Nature Rev. Neurosci. **7** (2006) 563-574
12. Gopych, P.M.: ROC Curves within the Framework of Neural Network Assembly Memory Model: Some Analytic Results. Int. J. Inf. Theo. Appl. **10** (2003) 189-197
13. Gopych, P.M.: Sensitivity and Bias within the Binary Signal Detection Theory, BSDT. Int. J. Inf. Theo. Appl. **11** (2004) 318-328
14. Gopych, P.M.: Neural Network Computations with Negative Triggering Thresholds. In: Duch, W. et al. (eds.): Artificial Neural Networks: Biological Inspirations. Lecture Notes in Computer Sciences Vol. 3696. Springer-Verlag, Berlin-Heidelberg (2005) 223-228
15. Gopych, P.M.: Generalization by Computation through Memory. Int. J. Inf. Theo. Appl. **13** (2006) 145-157
16. Gopych, P.M.: Performance of BSDT Decoding Algorithms Based on Locally Damaged Neural Networks. In: Corchado E. et al. (eds.): Intelligent Data Engineering and Automated Learning. Lecture Notes in Computer Sciences Vol. 4224. Springer-Verlag, Berlin-Heidelberg (2006) 199-206
17. Gopych, P.M.: Foundations of the Neural Network Assembly Memory Model. In: Shannon, S. (ed.): Leading-Edge Computer Sciences. Nova Sci., New York (2006) 21-84
18. Dayan, P., Hinton, P.E., Neal, R.M., Zemel, R.: The Helmholtz Machine. Neural Computations **7** (1995) 889-904
19. Gopych P.M.: Identification of Peaks in Line Spectra Using the Algorithm Imitating the Neural Network Operation. Instr. Experim. Techn. **41** (1998) 341-346
20. Mayfield, J.E.: Minimal History, a Theory of Plausible Explanation. Complexity **12** (2007) 48-53
21. Martinez-Conde, S., Macknik, S.L., Hubel, D.H.: The Role of Fixational Eye Movements in Visual Perception. Nature Rev. Neurosci. **5** (2004) 229-240