Reliable Computation in the Presence of Noise

S. Winograd and *J. D. Cowan*

The M.I.T. Press
Massachusetts Institute of Technology
Cambridge, Massachusetts

Library of Congress Catalog Card Number: 63-21195
Printed in the United States of America

4. The Reliability of Automata

4.1. Von Neumann's Solution

Perhaps the first to consider this problem in detail was von Neumann (1952, 1956). Pitts and McCulloch (1947), in attempting to apply the theory of automata to problems concerning auditory and visual perception and cortical functioning, had noted that the modular nets designed to model these aspects of complex biological systems ought to be constructed so that their function was unperturbed by small fluctuations in excitations and inhibitions, in thresholds of modules and in local details of connection between modules. Wiener (1948) had also considered an aspect of this problem, that of computer malfunctions caused by malfunctions and failures of their constituent modules. He had noted that in existing computers the use of majority logic—"What I tell you two out of three times is true"—coupled with a search procedure for finding new modules, resulted in reliability of computation.

It was with this background in mind that von Neumann attacked these problems. His principal solution to this problem was obtained in the following manner. It is desired to obtain a reliable output from a finite automaton, corresponding to some prescribed Boolian function. It is assumed that modules are available for the construction of the automaton, that compute one of two functions, either the Sheffer-stroke function $s(x_1, x_2)$, or else the majority function $m(x_1, x_2, x_3)$. Either all modules compute one of these functions, or else they all compute the other. In either case, arbitrary Boolian functions may be computed by nets of these modules, provided certain tricks are also utilized. Signals are transmitted from one module to another via connections that carry only binary impulses.

It is also assumed that modules fail to function correctly, with a (precise) probability ϵ. This malfunction is assumed to occur statistically independently of the general state of the net, and of the occurrence of other malfunctions. (In making this particular assumption, von Neumann observed that a more realistic one would be that

> the malfunctions are statistically dependent on the general state of the network, and of the occurrence of other malfunctions. In any particular state, however, a malfunction of the basic module in question has a probability of malfunction which is ϵ

but for ease of analysis, he adopted the simpler assumption.)

To minimize the effects of module malfunctions, a network is designed that comprises many more modules and connections than are absolutely necessary to compute the desired Boolian function; i.e., a *redundant* network is designed. Thus the finite automaton depicted in Figure 4.1,

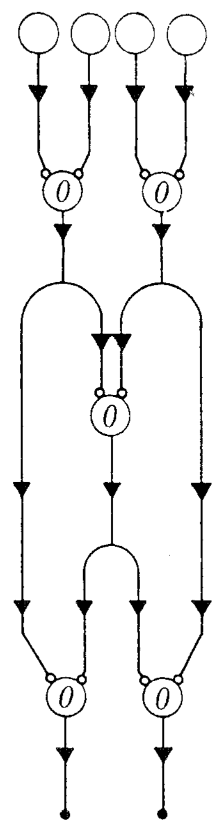

Figure 4.1. Irredundant modular net comprising only modules computing the Sheffer stroke function.

which realizes a given definite event, in a supposedly irredundant way, is used as a precursor for the redundant automaton depicted in Figure 4.2. This network has the following structure. Each module of the precursor network is replaced by an aggregate of modules of the same type, and

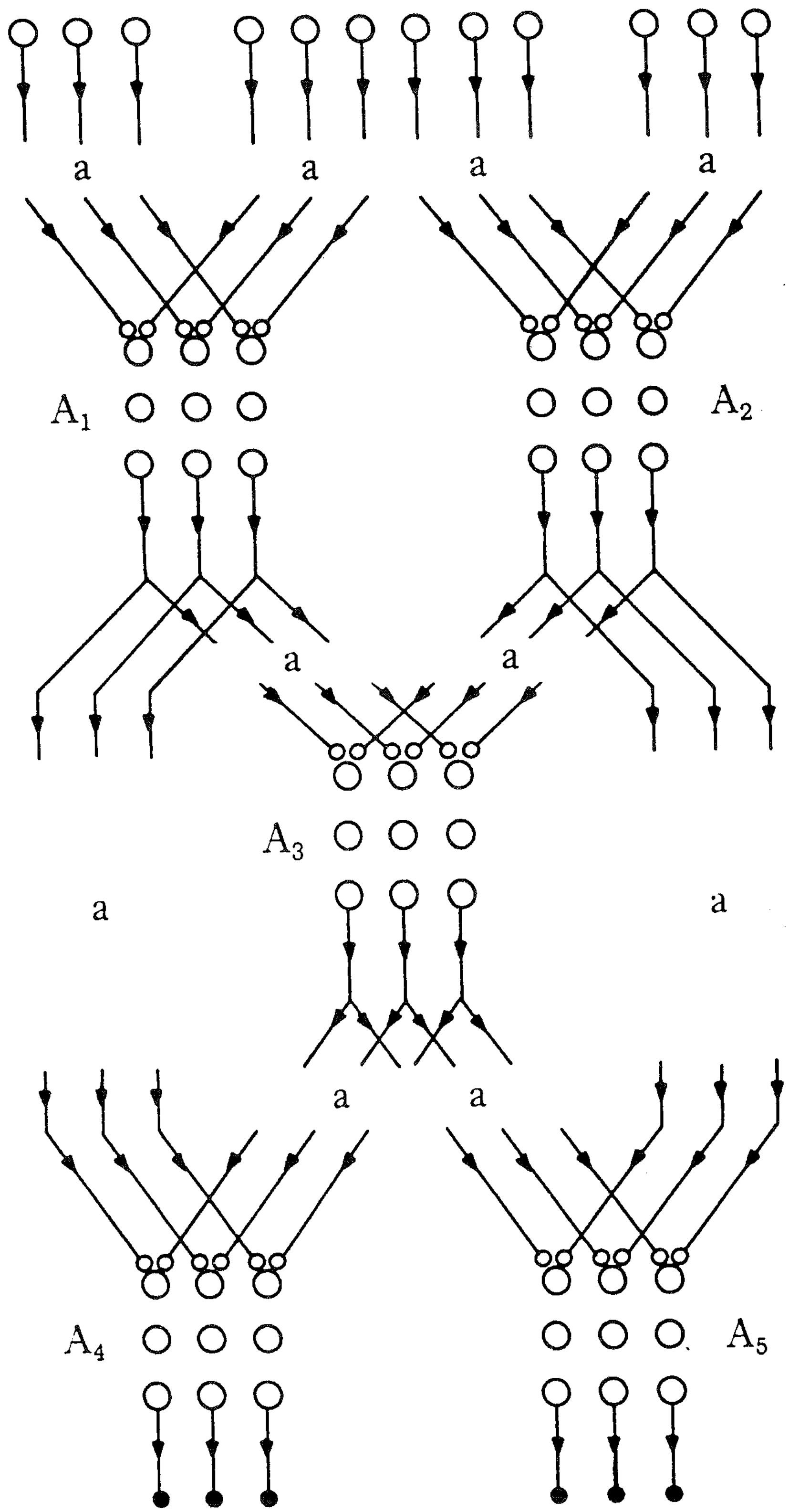

Figure 4.2. Redundant version of the network shown in Figure 4.1. A_r, redundant aggregates; *a*, permutations of 123.

each connection is replaced by a bundle. The organization within each aggregate and bundle is not completely ordered—some microlevel randomness is permitted. However, the organization between aggregates follows exactly that of the precursor network—no macrolevel randomness is permitted. Each aggregate and bundle processes only that which the precursor single module and connection processed. Thus many modules and connections of the network are redundant. It is this redundancy that allows the degrading effects of malfunctioning modules to be reduced, since each aggregate operates on a repeated signal carried by the bundle and makes a decision only on a majority basis.

The details of this process are as follows. Each bundle comprises n connections, each of which carries only binary impulses. There are thus 2^n distinct patterns of excitation or signal configurations per bundle, ranging from $(111\cdots1)$ to $(000\cdots0)$. The number of ones in any configuration is termed ξ, and a fiduciary level Δ is set, such that

$$\left.\begin{array}{lll} 1. & n \geqslant \xi \geqslant (1-\Delta)n & \text{signals } \mathit{1} \\ 2. & 0 \leqslant \xi \leqslant \Delta n & \text{signals } \mathit{0} \\ 3. & \Delta n < \xi < (1-\Delta)n & \text{signals malfunction} \end{array}\right\} \qquad (4.1)$$

Each redundant aggregate operates on its input bundles in the following manner.

Its first rank of modules computes n copies of $s(x_1, x_2)$; i.e., this rank computes $s(x_{1a_1}, x_{2b_1}), s(x_{1a_2}, x_{2b_2}), \cdots, s(x_{1a_n}, x_{2b_n})$, where $(a_1, a_2, \cdots, a_n)$ and $(b_1, b_2, \cdots, b_n)$ are any distinct permutations of $(1, 2, \cdots, n)$. The output bundle of this rank is then split, permuted, and its signal configurations are then processed by a second rank of modules. This second process is then iterated, and the resultant bundle carries signal configurations to other aggregates. The first rank is called, following von Neumann, an *executive organ* since it "executes" the given function of the aggregate. The set of succeeding ranks is called, a *restoring organ*, since its effect, as we shall describe, is to combat the degrading effects of module malfunctions occurring in the executive rank, and so restore signal configurations that are corrupted by noise. (Figure 4.3.) This is effected as follows: any signal configuration whose excitation level ξ, is greater than a certain initial value ξ_c has this level increased by the restoring organ, and conversely any signal configuration whose excitation level ξ is lower than ξ_c has its level decreased. (Figure 4.4.) By this means, malfunctions, which are represented by excitation levels taking values between n and $(1-\Delta)n$, are gradually corrected, since these particular excitation levels are gradually replaced by ones taking values outside the above region.

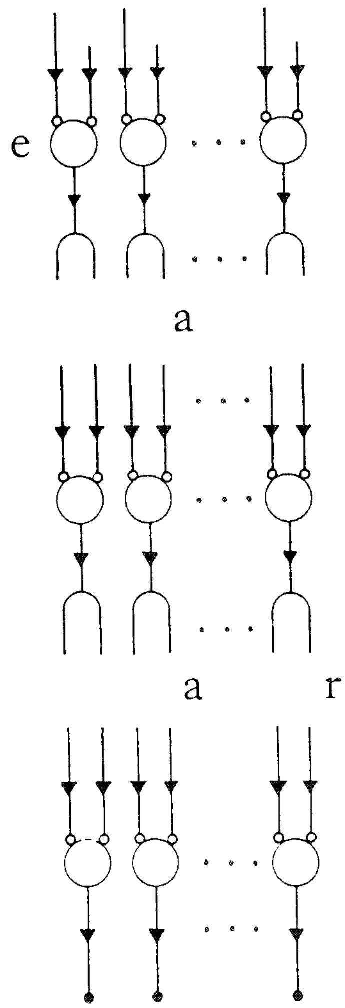

Figure 4.3. Redundant aggregate shown in detail. e, "executive" organ; r, "restoring" organ.

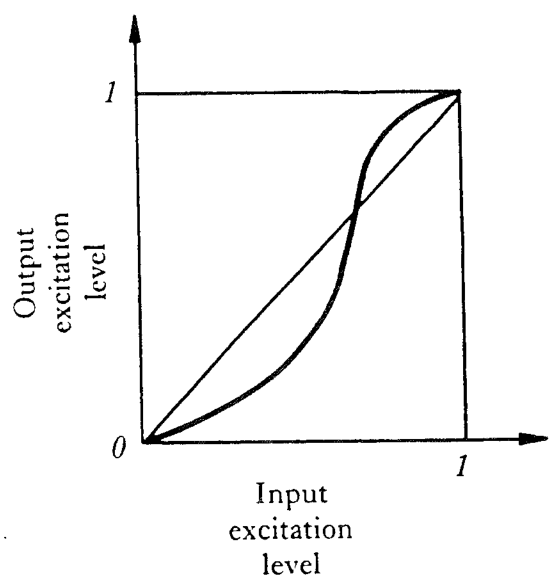

Figure 4.4. Transfer function of the restoring organ.

The aggregate itself is, as we have noted, redundant, and this redundancy may be defined as

$$N = 3n \tag{4.2}$$

For given probability of malfunction ϵ, von Neumann was able, by varying Δ and n, to show that an optimum Δ existed, such that no malfunctions propagated throughout the redundant network. He obtained for the over-all probability of malfunction of the network P, given modules that malfunction with a probability of error $\epsilon = 5 \times 10^{-3}$, the function

$$P \cong \frac{6.4}{n} \cdot 10^{-8.0n/10\,000} \tag{4.3}$$

This expression is such, however, that rather large values of n were required to obtain reasonably small P's. (See Table 4.1.) Von Neumann

TABLE 4.1. PROBABILITY OF NETWORK MALFUNCTION VERSUS REDUNDANCY

n	P
1 000	2.7×10^{-2}
2 000	2.6×10^{-3}
3 000	2.5×10^{-4}
5 000	4.0×10^{-6}
10 000	1.6×10^{-10}
20 000	2.8×10^{-19}
25 000	1.2×10^{-23}

was not satisfied with this solution and considered that

> error should be treated by thermodynamic methods, and be the subject of a thermodynamical theory, as information has been, by the work of L. Szilard and C. E. Shannon.

4.2. Other Approaches to the Design of Reliable Automata

Other approaches to the design of reliable automata involving the use of more complex modules with somewhat different error behavior than those previously cited also resulted in redundant automata with the same general character as that given by Equation 4.3. Thus Allanson (*op. cit.*) considered the design of networks comprising modules that compute a majority function of many inputs. A typical module is shown in Figure 4.5.

Errors are assumed to occur during synaptic transmission, i.e., in the transmission of signals between modules, as well as in modules themselves. These synaptic errors are assumed to be of two kinds: those that block an input signal, and those that spontaneously produce a fictitious input signal. We may conveniently represent the error behavior of a single

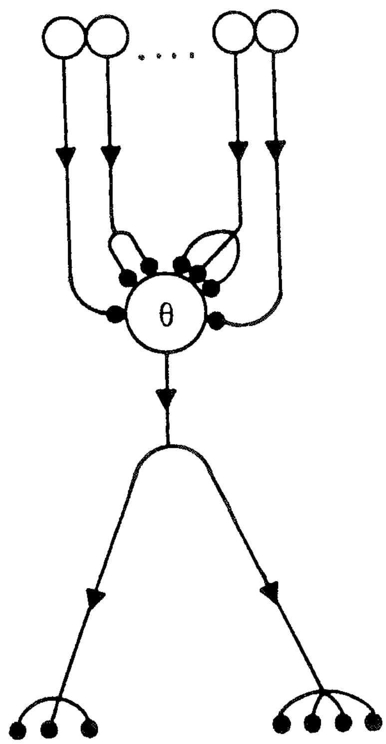

Figure 4.5. Complex module with synaptic replication.

synapse via the terminology of information theory, as a binary asymmetric channel, in which p_1 and p_2 are the respective "positive" and "negative" synaptic errors. Allanson found that an increase in the reliability of synaptic transmission could be obtained by increasing the number of endbulbs synapsing on any module, i.e., by synaptic replication.

The extension of these considerations to networks with both noisy modules and noisy connections was first considered by Muroga (*op. cit.*) and by Verbeek, Blum, Cowan, and McCulloch (*op. cit.*). They used modules similar to that described in Figure 4.6, except that with each module was associated a probability of error ϵ. In addition, synaptic errors occurring with probability p_s were considered. These errors were controlled, essentially by combining modular iteration and synaptic replication. For example, the function $P(t) \equiv \bar{y}_1(t-1) \vee \bar{y}_2(t-1)$ was computed reliably by the network shown in Figure 4.6. Each variable is represented by a bundle of n lines as before, and an excitation level $\xi: \Delta n < \xi < (1-\Delta)n$ signals a malfunction. The threshold θ of each of the n modules is set such that input errors are corrected, and only module errors remain. The condition for this is that

$$-2n(1-\Delta) < \theta \leqq -n(1+\Delta) \tag{4.4}$$

which leads to the equation

$$\theta = -(n + [n\Delta] + 0.5) \tag{4.5}$$

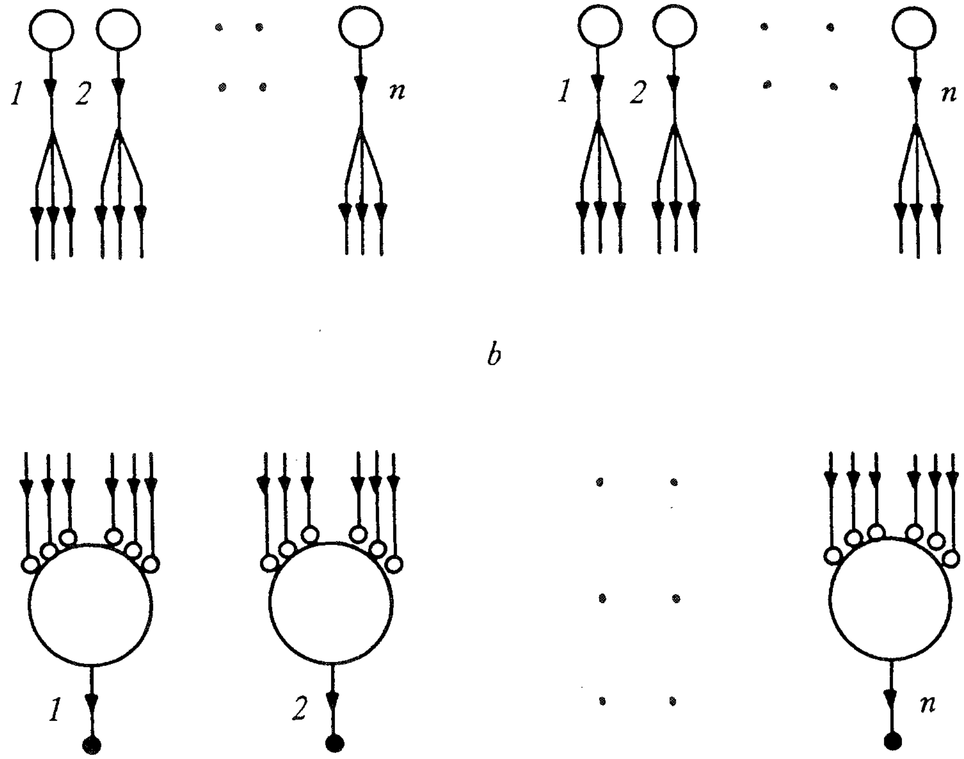

Figure 4.6. Error-controlling modular network. *b*, All-to-all connection pattern.

where $\Delta < 0.33$ and $[n\Delta]$ is the smallest integer greater than $n\Delta$. We note that each input line synapses on all n modules, each of which computes the threshold function given by Equation 4.5.

An upper bound on the probability of network malfunction owing to input and synaptic errors may be calculated as a cumulative binomial probability distribution. Combining this with the probability of network malfunction owing to module errors provides an upper bound to the probability of malfunction for the output bundle. From this the cumulative binomial probability of malfunction P of the network can be calculated. This calculation is complicated by the fact that input and synaptic errors may or may not produce independent effects in the output. An upper bound on P may be obtained, however, by calculating the probability of network malfunction for both cases, and taking the resultant largest value of the two for P. The results are as follows: Let ϵ_1 be the probability of input errors (in the example of Figure 4.6 this would be equivalent to assuming that the input modules malfunction with probability ϵ_1). Then

$$\eta = \epsilon_1(1 - p_s) + (1 - \epsilon_1)p_s \tag{4.6}$$

is the probability of malfunction due to the effects of both input and synaptic errors. Table 4.2 shows the variation of P with n, η, and ϵ, in

TABLE 4.2. PROBABILITY OF MODULAR MALFUNCTION VERSUS REDUNDANCY

	max P	
n	$\eta = \varepsilon = 0.005$	$\eta = \varepsilon = 0.01$
5	6.7×10^{-2}	2.4×10^{-1}
10	3.1×10^{-3}	3.8×10^{-2}
20	1.0×10^{-4}	7.2×10^{-3}
30	3.6×10^{-6}	1.5×10^{-3}
40	$\sim 10^{-7}$	5.6×10^{-5}

the case that $\eta = \epsilon$. Comparison of Table 4.2 with Table 4.1 provides some indication of the increased efficiency obtained by using more complex modules to control malfunctions. In fact, the increased efficiency is such that errors are controlled within networks of unit depth and error-restoring networks are not required. Alternatively, the more complex modules may be considered to realize a combination of computation and error-restoring, all in one location (Cowan, 1960*b*). However, the probability of malfunction P goes to zero only as n goes to infinity, similarly to von Neumann's system: except that the approach to infinity is slower for given P. Thus, although an increased efficiency compared with von Neumann's construction has been obtained, arbitrarily high redundancy is still required to achieve arbitrarily low probabilities of malfunction.

4.3. A Comparison with Information Theory

If we compare the redundant modular networks designed by von Neumann *et al.* with the redundant communication system of Shannon (1948, *op. cit.*), certain fundamental differences are immediately apparent. We shall compare modular redundancies with that redundancy of signal sequence used to combat the effects of noisy channels. We define the *modular redundancy* of a network, N, to be the ratio of the number of modules in the redundant network to the number of modules in the irredundant one. Thus

$$\left.\begin{aligned} N &= 3n \quad \text{(von Neumann)} \\ N &= n \quad \text{(Muroga } et\ al.\text{)} \end{aligned}\right\} \tag{4.7}$$

Similarly, the *computation ratio* per module R is defined to be $1/N$ so that

$$\left.\begin{aligned} R &= \frac{1}{3n} \quad \text{(von Neumann)} \\ R &= \frac{1}{n} \quad \text{(Muroga } et\ al.\text{)} \end{aligned}\right\} \tag{4.8}$$

It is evident from Tables 3.3 and 3.4 that the probability of malfunction P is approximately proportional to $\exp(-c/R_a)$; that is,

$$
\left.
\begin{aligned}
P &\cong \mathrm{d}_1 \cdot \exp\left(-\frac{\mathrm{c}_1}{R}\right) \quad \text{(von Neumann)} \\
P &\cong \mathrm{d}_2 \cdot \exp\left(-\frac{\mathrm{c}_2}{R}\right) \quad \text{(Muroga \textit{et al.})}
\end{aligned}
\right\} \tag{4.9}
$$

where d_1 and d_2 are slowly varying functions of n, and where c_1 and c_2 are constants.

If we compare Equation 4.9 with the expression for the probability of identification errors, P derived from Equation 3.15, i.e.:

$$
\mathrm{P} \cong 2^{-\mathrm{n}(\mathrm{C}-\mathrm{R})} \tag{4.10}
$$

we see an obvious difference in that P goes to zero with n, independently of R, provided only that $\mathrm{R} \leqq \mathrm{C}$, whereas P goes to zero only with R. This situation is depicted in Figure 4.7:

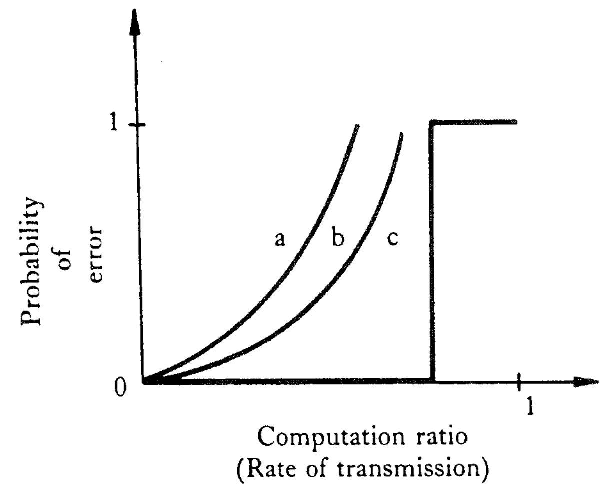

Figure 4.7. Comparison of different solutions to the reliability problem. *a*, von Neumann; *b*, Muroga *et al.*; *c*, Shannon.

The import of this is that the redundant automata designed by von Neumann and by Muroga *et al.* do not exhibit noise-free behavior of the type that would permit computation capacities to be defined.

4.4. Attempts to Apply Information Theory

The absence of computation capacities for the redundant automata considered in Sections 4.1 and 4.2 suggested that more direct applications of coding theory to the computation problem might result in redundant automata displaying such capacities. Elias (*op. cit.*) accordingly set up a model of a computation system consisting of a noisy modular network serviced by noiseless encoders and decoders. The structure of this system, which we have modified somewhat without changing its essential features, is depicted in Figure 4.8. The modular network is required to realize the event E that corresponds to a logical function of the inputs x_1 and x_2. Elements of the network comprise modules that compute

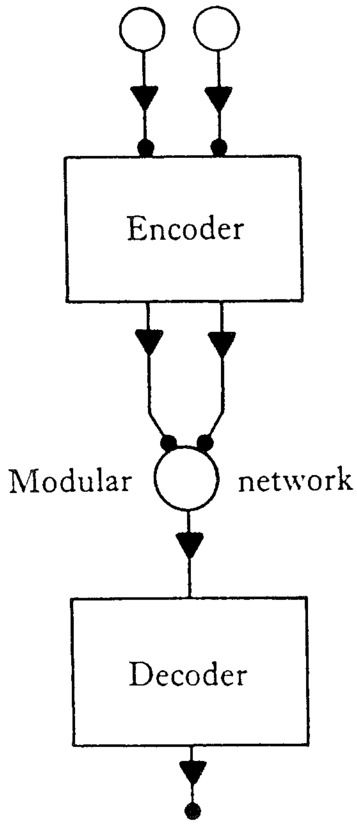

Figure 4.8. Computation system.

Boolian functions of one or two variables, e.g., $\bar{}$, $\&$, v, $\equiv$, $\oplus$, etc., with small but nonzero probabilities of error. These probabilities are statistically independent of the general state of the network, and of the occurrence of other malfunctions; i.e., the same error assumption as von Neumann's is made. The modular network is, however, irredundant, and the redundancy in the system is not of hardware or channel but is simply redundancy of signal sequence, as in the case of communication. Sequences of length k of inputs x_1 and x_2 are encoded into signal sequences of length n, and these serve as inputs to the network, and the output sequences of length n are decoded into sequences of length k, which, one hopes, correspond to the event to be realized. It is important to note, however, that the network is memoryless, and all processing occurs digit by digit, so that no operation occurs within a sequence.

It is assumed that each input x_1 and x_2 is independently encoded, and that if the modular network were to be noiseless, then the decoder maps signal sequences 1:1 into output sequences. These assumptions ensure that the prescribed set of events be realized entirely in the noisy modular network and not (partly or wholly) in the noiseless encoding or decoding networks.

The results obtained were as follows. Of the sixteen possible Boolian functions of two inputs x_1 and x_2, only eight were such that (n, k) codes could be used that realized positive rates of transmission of information through the network at vanishing small frequencies of error. Of these eight, six were of little interest, being the functions I (tautology), O (contradiction), x_1, x_2, $\bar{x}_1$, and $\bar{x}_2$. The remaining two, $x_1 \oplus x_2$ and $x_1 \equiv x_2$, are such that each maps the set of 2^k n-digit sequences into itself in such a way that the metrical properties of the set are preserved, i.e., sequences differing from one another by a large number of digits, in the input map into similar sequences in the output. This means that Hamming codes (*op. cit.*) or any other group codes (Slepian, 1956) may be used to code input sequences to these functions. However, $x_1 \oplus x_2$ and $x_1 \equiv x_2$ are not universal functions, and so the entire set of eight functions is incomplete; i.e., networks that realize even definite events cannot in general be constructed from only members of this set.

Elias showed that for the other set of eight functions, comprising $x_1 \,\&\, x_2$, $x_1 \,\&\, \bar{x}_2$, $\bar{x}_1 \,\&\, x_2$, $\bar{x}_1 \,\&\, \bar{x}_2$, $x_1 \vee x_2$, $x_1 \vee \bar{x}_2$, $\bar{x}_1 \vee x_2$, and $\bar{x}_1 \vee \bar{x}_2$, no code is better than that one which has the digit encoders repeat each input digit n times and takes a majority of the outputs as the correct output. This implies that for those eight functions, arbitrarily high reliability of computation can be achieved only at the expense of arbitrarily low rates of computation. Elias' analysis was repeated by Peterson and Rabin (*op. cit.*), who obtained substantially the same result.

4.5. Discussion

We may conveniently separate the considerations reported so far into a number of different classes. The first class consists of those considerations of von Neumann in which essentially simple modules are studied that malfunction with probability ε. Redundant aggregates and sequences consist essentially of many copies of simple modules, or else of message digits. That is, in the process of coding no increase of modular complexity occurs. (We use as a measure of modular complexity the number of inputs to a module.) The resultant networks certainly control and correct errors, but do so, generally, in an inefficient manner. More efficient constructions were discovered by Allanson, Muroga, and Verbeek *et al.* (*op. cit.*) wherein aggregates of complex modules were used to replace simple modules.

In none of these classes is any positive rate for reliable information processing exhibited, in the sense that we have previously defined. It is evident, however, that increased efficiency in the coding of redundant automata may be obtained at a cost of increased complexity of the modules comprising such automata, leading to the conclusion that only automata consisting of redundant aggregates of complex modules can exhibit nonvanishing computation ratios for reliable modular computation.