

RELIABLE NETWORKS FROM UNRELIABLE GATES WITH ALMOST MINIMAL COMPLEXITY

Dietmar Uhlig
IH Mittweida, Platz der DSF 17
Mittweida, 9250, DDR

ABSTRACT: We consider (combinatorial) networks constructed by using unreliable gates with a given error probability. We show that for almost all Boolean functions f there are networks realizing f , having almost the same error probability as the gates and having almost the same complexity as the minimal (unreliable) networks realizing f in case no gate has failed (having a very great error probability). This may be contrasted with results of 1.) von Neumann (1952), 2.) Dobrushin/Ortyukov (1977), 3.) Pippenger (1985) to the effect that the number of gates needed 1.) for minimal (reliable) networks is larger by at most a logarithmic factor than the number needed for unreliable networks [5], 2.) for some Boolean functions is larger by at least a logarithmic factor, 3.) for almost all Boolean functions is a (very great) multiple of the number of gates for unreliable realizations.

1. Definitions and Dobrushin, Ortyukov and Pippenger's main results

To describe the main results of other authors in this field and to formulate and sketch our theorems we need several definitions and notations. Precise definitions of the (combinatorial) networks are given in [4,8]. The networks are constructed by gates γ from a complete set \mathcal{G} of gates. For example, the set consisting of 2-input AND, 2-input OR and the NOT function is complete. A cost or weight (a positive number), $C(\gamma)$, is associated with each of the gates $\gamma \in \mathcal{G}$. The cost $C(D)$ of a network D is the sum of costs of its gates. Let f be an arbitrary Boolean function. We set $C(f) = \min C(D)$, where D ranges over all networks realizing the function f assuming no gate has failed. Let us further suppose that each of the gates of the set \mathcal{G} has an error probability ϵ ($\epsilon < 1/2$), i.e. the probability of the event "the gate does not realize the function according to it" is equal to ϵ . Let $\tilde{a} = (a_1, \dots, a_n)$ be any input vector of a network D . We define the

error probability $p_{\tilde{a}}(D)$ according to input vector \tilde{a} and to network D as the probability that network D for input vector \tilde{a} does not compute the value which it computes in case no gate has failed. The error probability $p(D)$ of a network D is defined as $\max p_{\tilde{a}}(D)$, where \tilde{a} ranges over all input vectors of network D . We set $C_{\delta}(f) = \min C(D)$, where D ranges over all networks realizing function f provided no gate has failed and having error probabilities not greater than δ . Furthermore, we define $C(n) = \max C(f)$, $C_{\delta}(n) = \max C_{\delta}(f)$, where f ranges over all n -argument Boolean functions.

We will use the following notations:

$a(n) \sim b(n)$ denotes $a(n) = b(n)(1 + \delta_n)$, where $\delta_n \rightarrow 0$ if $n \rightarrow \infty$,
 $a(n) = o(b(n))$ denotes $a(n) = \delta_n b(n)$, where $\delta_n \rightarrow 0$ if $n \rightarrow \infty$,
 $a(n) \lesssim b(n)$ denotes that there is a constant c_1 such that $a(n) \leq c_1 b(n)$,
 $a(n) \lesssim b(n)$ denotes $a(n) \leq b(n)(1 + \delta_n)$, where $\delta_n \rightarrow 0$ if $n \rightarrow \infty$.
 Further, \log_a denotes $\log_2 a$ and $\ln a$ denotes $\log_e a$.

If \mathcal{G} is an arbitrary element of the given complete set, then we consider number $\mathcal{G}(\mathcal{G})$ defined as costs of elements divided by $v(\mathcal{G}) - 1$, where $v(\mathcal{G})$ is the number of inputs of \mathcal{G} . Let \mathcal{G} be the minimum of all those numbers. We call the element for which we get this minimum the "cheapest" element of set \mathcal{G} . Denote by Q_n the set of all Boolean functions of arguments x_1, \dots, x_n and by Q_{n, c_2} its subset of all Boolean functions with complexity $C(f) \leq \mathcal{G}(1 - c_2)2^n/n$. A well-known result of O.B. Lupanov [4] is the following (proved in 1958):

1. $C(n) \sim \mathcal{G} 2^n/n$. (1)
2. For each constant c_2 , $c_2 \in (0, 1)$ $|Q_{n, c_2}| = o(|Q_n|)$

(i.e. "almost all" functions have the complexity of the function with the highest complexity).

For the reader's convenience let us assume that the set \mathcal{G} contains a gate \mathcal{G} realizing the 3-argument majority function $xy \vee xz \vee yz$ (having also error probability ϵ). First assume $\epsilon = \text{const}$. R.L. Dobrushin and S.I. Ortyukov have shown in [1] if $\epsilon < 0.07$ and $\delta \gg q(\epsilon)$ where $q(\epsilon)$ is the minimum of the positive roots of the expression $\epsilon + (1 - \epsilon)(3q^2 - 2q^3) = q$ then there is a constant $c_3 = c_3(\epsilon)$ such that for each network D there is a network \tilde{D} realizing the same function as (provided no gate of D has failed), having error probability $p(\tilde{D}) \leq \delta$ and complexity

$$C(\tilde{D}) \lesssim c_3 C(D) \ln C(D) \quad \text{if } C(D) \rightarrow \infty. \quad (2)$$

Dobrushin and Ortyukov have also shown [2] that there are an infinite

number of Boolean functions f and a positive number c_4 such that

$$C(f) \geq c_4 C(f) \ln C(f).$$

N. Pippenger's main result [7] can be described as follows. There exists a constant c_5 (it seems to be very great) such that for every f , $f \in Q_n$,

$$C_{\delta}(f) \leq c_5 \xi 2^n / n$$

(i.e., for almost all Boolean functions the complexities of the reliable realizations are only a multiple of the complexities of the unreliable realizations).

S.I. Ortyukov [6] also considered error probabilities ϵ_n with $\epsilon_n \rightarrow 0$ if $n \rightarrow \infty$. Let $\delta = \delta_n$, where $\epsilon_n \in (0, 1/2)$ and

$$\overline{\lim} n^{-1} \log \epsilon_n \leq -1. \quad (3)$$

Then for each constant $c_6 > 0$ if $\delta_n \geq (1+c_6)\epsilon_n$ then

$$C_{\delta_n}(n) \sim C(n) \sim \xi 2^n / n.$$

But (3) is a very strong and artificial restriction.

2. New results, sketch of proof

A few weeks ago the author of this paper prepared a publication for the proceedings of the "International Workshop on Parallel Algorithms and Architectures" [9] to be held in Suhl with a sketch of proof of a theorem being called Theorem 2 in the presented paper. This Theorem 2 is a corollary of the following

Theorem 1. For each arbitrarily small positive constant γ and any two sequences $\epsilon_n \in (0, 1/2)$ and $\delta_n \in (0, 1/2)$ where

$$\begin{aligned} \epsilon_n &\rightarrow 0, \\ \delta_n &\geq (1+\gamma)\epsilon_n \end{aligned}$$

if each network realizing a Boolean function of n variables consists only of elements with an error probability not greater than ϵ_n then

$$C_{\delta_n}(n) \sim C(n) \sim \xi 2^n/n$$

(i.e., for almost all Boolean functions the complexities of the reliable realizations are almost the same as the complexities of the unreliable realizations).

Theorem 2. For any arbitrarily small positive numbers c and γ there exists a number ϵ' , where $\epsilon' \in (0, 1/2)$, with the property that if $0 < \epsilon \leq \epsilon'$, if each element of the given complete set \mathcal{G} has an error probability not greater than ϵ and if $\delta \geq (1+\gamma)\epsilon$ (more precisely, if $\delta \geq q(\epsilon)$ (see Dobrushin/Ortyukov's Theorem [1])) then

$$C_{\delta}(n) \lesssim (1+c)\xi 2^n/n \sim (1+c)C(n).$$

Note. Suppose the complete set \mathcal{G} does not contain an element realizing function $xy \vee xz \vee yz$. Then Theorem 1 and 2 would remain valid if we take δ (respectively δ_n) as follows:

$\delta \geq (1+\gamma)\xi\epsilon$ ($\delta_n \geq (1+\gamma)\xi\epsilon_n$), where ξ is the minimal number of elements which are sufficient to realize this function.

The method of realization of Boolean functions satisfying Theorem 1 is almost the same as for functions satisfying Theorem 2, but some of the parameters are chosen in another way and the estimation of the error probability of the networks is more difficult. In our method we shall use modifications of the methods of other authors, namely Kirienko's construction of so-called self-correcting networks [3], Ortyukov's idea of construction and estimation of probabilities [6], Lupanov's general method of realisation of Boolean functions [4] and Dobrushin/Ortyukov's method [1].

Let y be an arbitrary Boolean variable or Boolean function. Denote $K_0(y) = \bar{y}$, $K_1(y) = y$. If $\vec{\sigma} = (\sigma_1, \dots, \sigma_u)$ is an arbitrary Boolean vector, then by $K_{\vec{\sigma}}(y_1, \dots, y_u)$ we denote the conjunction $K_{\sigma_1}(y_1) \dots K_{\sigma_u}(y_u)$ and by $|\vec{\sigma}|$ denote $\sigma_u 2^{u-1} + \sigma_{u-1} 2^{u-2} + \dots + \sigma_1$.

If we have to compute the function $f(x_1, \dots, x_n)$ we will represent it by

$$f(x_1, \dots, x_n) = \bigvee_{\vec{\alpha}} K_{\vec{\alpha}}(x_{m+1}, \dots, x_n) f(x_1, \dots, x_m, \alpha_{m+1}, \dots, \alpha_n), \quad (4)$$

where $\vec{\alpha}$ denotes Boolean vector $(\alpha_{m+1}, \dots, \alpha_n)$. Let us take Boolean functions g_1, \dots, g_L of m variables such that for every Boolean vector $\vec{a}' = (a_1, \dots, a_m)$ the vector $\vec{g}(\vec{a}') = (g_1(\vec{a}'), \dots, g_L(\vec{a}'))$ is a BCH code vector of vector

$$\vec{f}(\vec{a}') = (f(\vec{a}', 0, \dots, 0), \dots, f(\vec{a}', \alpha_{m+1}, \dots, \alpha_n), \dots, f(\vec{a}', 1, \dots, 1))$$

correcting $R = R(L)$ errors, where L , R and m are parameters depending on n in such a manner that

$$L \sim 2^{n-m}, n \sim m \quad (5)$$

and that R is sufficiently large. The network A realizing function f consists of subnetworks A_{11}, \dots, A_{1L} and \tilde{A}_2 , where A_{11}, \dots, A_{1L} realize the functions g_1, \dots, g_L and where \tilde{A}_2 for each input vector $\tilde{a} = (a_1, \dots, a_n)$ according to the output vector of the networks A_{11}, \dots, A_{1L} (assuming at most R of them has failed and supposing \tilde{A}_2 itself computes correctly) will determine vector $\tilde{f}(\tilde{a}')$ and will put out $f(\tilde{a})$ to it.

Let us consider the event $E_R =$ "more than R of the networks A_{1j} , $j=1, \dots, L$, has failed". The networks A_{1j} are constructed in such a way that the sum of complexities of all of them is

$$\sum_{j=1}^L C(A_{1j}) \sim \xi 2^n/n \quad (6)$$

and that

$$\Pr(E_R) \leq (\gamma/2)\epsilon_n, \quad (7)$$

(γ is the constant of Theorem 1.)

Network \tilde{A}_2 has an error probability not greater than $(1+\gamma/2)\epsilon_n$. Then by (7) we obtain that the error probability of network A satisfies Theorem 1. Expression (7) is satisfied if each network A_{1j} will be realized independently of A_{1l} , $l \neq j$, with error probability at most $(c_7 + N_2)\epsilon_n$, where c_7 is a sufficiently great constant and where N_2 sufficiently slowly tends to infinity. Function N_2 depends on n and is equal to the number of elements of a network later used in our construction.

For our construction we need a more precise inequality than (2) already used by Ortyukov [6] (It follows from [1]):

$$C(\tilde{D}) \leq c_3 C(D) \ln C(D) + O(C(D)) + O((\ln(1/\epsilon)) \log^3), \quad (8)$$

if $C(D) \rightarrow \infty$, $\epsilon \rightarrow 0$.

To construct network \tilde{A}_2 we firstly consider an other network A_2 realizing the same Boolean function as \tilde{A}_2 provided no gate has failed. (The error probability may be very great.) It consists of networks A_{21} and A_{22} . Network A_{21} puts out the vector $\tilde{f}(\tilde{a}')$ to BCH code vector $\tilde{g}(\tilde{a}')$. It is well-known that it can be realized with complexity $C(A_{21}) \leq L^4$. Network A_{22} realizes the Boolean function $K_{\tilde{a}}(x_{m+1}, \dots, x_n) \cdot y$.

Obviously, $C(A_{22}) = 2^{n-m}$. Thus,

$$C(A_2) = C(A_{21}) + C(A_{22}) \leq (\text{see (5)}) \leq 2^{4(n-m)}.$$

By this and by (5) and (8) follows $C(\tilde{A}_2) = o(2^n/n)$.

The construction of networks A_{1j} is very complicated.

Suppose that the reader is familiar with Lupanov's general method of realization of Boolean functions $f(x_1, \dots, x_n)$ with complexities given by (1).

Each of the A_{1j} realizes a Boolean function of m variables. It is constructed in such a way that its error probability is not greater than $(c_7 + N_2)\epsilon_n$ and that nevertheless the complexity is $\mathcal{O}(1+o(m))2^m/m$, from which follows (6) by (5). Network A_{1j} consists of a subnetwork \tilde{A}_{1j1} , of a lot of subnetworks A_{1j2} and of a subnetwork \tilde{A}_{1j3} . (Together with a short description of Lupanov's method these networks are described in a more detailed way in [9], but for the proof of the presented Theorem 1 the parameters must be chosen in an other way, especially $N_2 \rightarrow \infty$ instead of $N_2 = \text{const.}$) Instead of a network having many inputs and realizing a conjunction we take network \tilde{A}_{1j3} . This network has special properties. To each of its inputs there exists exactly one network A_{1j2} the output of which is connected with the corresponding input. If A_{1j} computes correctly, then only one of the networks A_{1j2} may put out the signal 1, if it does not compute correctly, then there may be more than one signal 1. Network \tilde{A}_{1j3} is constructed in such a way that it puts out the signal being exactly on one of its inputs with an error probability at most $2\epsilon_n$, that $C(\tilde{A}_{1j3}) = o(2^m/m)$ and that the selection of the input depends on the input vector of network A .

The "main" networks in Lupanov's construction consist only of "cheapest" elements which are connected in such a way that the number of inputs is a maximum. The networks A_{1j2} are our "main" networks. Each of them in turn consists of a network \tilde{A}_{1j22} with N_1 inputs and error probability not greater than $2\epsilon_n$ and consists of N_1 networks A_{1j21} being "main" networks in Lupanov's construction and having N_2 inputs. Network \tilde{A}_{1j22} is constructed in such a way that $C(\tilde{A}_{1j22}) = o(N_1 \cdot C(A_{1j21}))$ if $N_1 \rightarrow \infty$, $N_2 \rightarrow \infty$ and that for each input vector of A_{1j} any false input signal, in most cases, do not affect a false output signal. More precisely, if the input signals which are independent to each other are false with a probability of at most $4\epsilon_n$, then the probability of a false output signal is not greater than $(c_7 + N_2)\epsilon_n$ and expression (7) is satisfied. The network \tilde{A}_{1j1} contains a lot of subnetworks having an error probability of at most $4\epsilon_n$ and realizing Boolean functions analogous to the functions used by Lupanov's method.

For the complexity we have $C(\tilde{A}_{1j1}) = o(2^m/m)$.

3. Networks from unreliable and reliable gates

Now we will characterize so-called r -self-correcting (combinatorial) networks. Let \mathcal{G}_0 be a complete set of reliable gates. These networks consist of elements of \mathcal{G} and of \mathcal{G}_0 and are constructed in such a way that the failure of any r' ($r' \neq r$) of its elements taken from \mathcal{G} does not effect its correct operation.

Let $C^r(n)$ be defined for r -self-correcting networks analogous to $C(n)$. G.I. Kirienko has shown in [3] that if $r=2^{o(n)}$, then

$$C^r(n) \sim \varrho 2^n/n \quad (9)$$

($\varrho = \varrho(\mathcal{G})$), i.e. the costs of elements of \mathcal{G}_0 may be very high) and the author of this paper has shown in [10] that if \mathcal{G}_0 is a complete set networks containing $c_g r$ reliable gates of \mathcal{G}_0 (where c_g is a sufficiently large positive number) and satisfying (9) can be constructed and that, on the other hand, c_{gr} ($c_g > 0$) reliable gates of \mathcal{G}_0 are needed [11].

Using a small number of reliable gates (with error probability equal to 0, where the costs of the reliable gates may be very high) we obtain the following

Theorem 3. Suppose, additional to \mathcal{G} , there is a complete set of (reliable) gates \mathcal{G}_0 .

Then for any arbitrarily small positive numbers c and q there exists a number ϵ' , where $\epsilon' \in (0, 1/2)$, with the property that if $0 < \epsilon \leq \epsilon'$ and if each element of the given complete set \mathcal{G} has an error probability not greater than ϵ and if $\delta \geq p$ then

$$C_\delta(n) \lesssim (1+c)\varrho 2^n/n \sim (1+c)C(n).$$

References

- [1] Dobrushin, R.L. and S.I. Ortyukov: Upper Bound for Redundancy of Self-Correcting Arrangements of Unreliable Functional Elements. Prob. of Info Transm. 13 (1977), 203-218.

- [2] Dobrushin, R.L. and S.I. Ortyukov: On the Lower Bound for Redundancy of Self-Correcting Networks of Unreliable Functional Elements. *Prob. Peredači Informacii* 13 (1977) 1, 82-89. (Russian)
- [3] Kirienko, G.I.: Synthesis of Combinatorial Networks which are Self-Correcting Referring to a Growing Number of Errors. *Diskretnij analiz, Sb. Trudov IM SO AN SSSR* 16 (1970) 38-43. (Russian)
- [4] Lupanov, O.B.: On a Method of Synthesis of Networks. *Izv. Vysš. Učebn. Zavad. Radiofizika* 1 (1958) 1, 120-140. (Russian)
- [5] Neumann von, J.: Probabilistic Logic of Reliable Organism from Unreliable Components. In: C.E. Shannon and J. McCarthy (Eds.), *Automata studies*, Princeton University Press (1956) 43-98.
- [6] Ortyukov, S.I.: On the Synthesis of Asymptotically Nonredundant Self-Correcting Networks of Unreliable Functional Elements. *Prob. Peredači Informacii* 13 (1977) 4, 3-8. (Russian)
- [7] Pippenger, N.: On Networks of Noisy Gates 26. Symposium on Foundation on Computer science, 21. - 23. 10. 1985, Portland, 30-38.
- [8] Savage, J.E.: *The Complexity of Computing*. Wiley-Interscience, New York, 1976.
- [9] Uhlig, D.: On Reliable Networks from Unreliable Gates. Prepared for the Proceedings of the "International Workshop on Parallel Algorithms and Architectures" 25. - 30. 5. 1987, Suhl.
- [10] Uhlig, D.: Combinatorial Networks which are Self-Correcting and have almost the Smallest Complexity. *Wiss. Beitrage der FSU Jena, Kompliziertheit von Lern- und Erkennungsprozessen* (1975). 225-228. (German)
- [11] Uhlig, D.: On the necessary proportion of reliable elements, *Vortraege zur Automatentheorie, Weiterbildungszentrum fuer MKR der TU Dresden* 6 (1974) 72-76. (German)