# On the Maximum Tolerable Noise of $k$-input Gates for Reliable Computation by Formulas

William S. Evans *Member, IEEE* and Leonard J. Schulman

***Abstract*— We determine the precise threshold of component noise below which formulas composed of odd degree components can reliably compute all Boolean functions.**

***Index Terms*— Computation by unreliable components, reliable computing**

## I. Introduction

We consider a model of computation that was first proposed by von Neumann in 1952 [1]: the *noisy circuit*. A noisy circuit is composed of $\epsilon$-noisy gates. An $\epsilon$-noisy, $k$-input gate is designed to compute a Boolean function of its $k$ Boolean inputs; however, it has the property that for any assignment to the inputs, there is probability $\epsilon$ that the output of the gate is the complement of the designed value. In a noisy circuit this event occurs independently at every gate of the circuit.

A circuit takes $n$ Boolean values as input and produces one Boolean output. The inputs to a gate in the circuit may be the outputs of other gates in the circuit, inputs to the circuit, or the constants 0 or 1. The output of the circuit is the output of one of the gates called the top gate.

The interconnection pattern does not allow "feedback." That is, the interconnection structure of the circuit is a directed, acyclic graph: vertices of the graph correspond to gates, and a directed edge from $u$ to $v$ corresponds to gate $v$ taking as input the output of gate $u$. If, in addition, the graph forms a tree (each vertex having one outgoing edge) then we call the circuit a *formula*.

Clearly, a noisy circuit with $\epsilon > 0$ cannot deterministically compute a Boolean function $f$; on any input there is probability at least $\epsilon$ that the top gate will output the complement of $f$. (We assume without loss of generality that $\epsilon \leq 1/2$.) The *error probability* of a noisy circuit for a Boolean function $f$ is the maximum over all inputs of the probability that the circuit's output differs from the value of the function. If this maximum is at most $\delta$ then the circuit $(1-\delta)$-*reliably* computes the function.

Fixing $k$, we are interested in the maximum value of $\epsilon$ for which it is possible to have *reliable computation*, which we define as: there is a $\delta < 1/2$ so that for every Boolean function there exists a noisy circuit using arbitrary $\epsilon$-noisy, $k$-input gates, that $(1-\delta)$-reliably computes the function. The word *reliable* in this context does not mean perfectly accurate, but rather that the output of the noisy circuit is biased, by a fixed amount, towards the correct output on every input.

The need for a limit on gate noise in order to achieve reliable computation was first noticed by von Neumann, who showed that for $\epsilon < 0.0073$, reliable computation is possible using $\epsilon$-noisy, 3-input majority gates. His method was to interleave "computation levels" of the circuit, i.e., levels that correspond to levels of the original (noiseless) circuit, with "error-correction levels," in which 3-input majority gates combine the output of three separate copies of each computation, in order to obtain an output that is more likely to be correct than any single copy.

As von Neumann noted, this idea cannot lead to reliable computation if $\epsilon \geq 1/6$. Consider a particular 3-input majority gate. If each of its inputs is incorrect independently with probability $a$, then it in turn will be incorrect with probability

$$(1-\epsilon)(a^3 + 3a^2(1-a)) + \epsilon((1-a)^3 + 3a(1-a)^2). \quad (1)$$

If $\epsilon < 1/6$ then this value can be smaller than $a$; this amplification is necessary for von Neumann's argument to work. However, if $\epsilon \geq 1/6$, then for every $a < 1/2$, the error probability of the output is greater than $a$, i.e., the output of the majority gate is less reliable than its inputs, and von Neumann's method fails.

This suggested to von Neumann that perhaps reliable computation is not possible by $\epsilon$-noisy, 3-input gates if $\epsilon \geq 1/6$. The first proof that there is some $\epsilon < 1/2$ for which reliable computation by noisy components is impossible, came in 1988 from Pippenger's work on formula depth[1] bounds [2]. He proved that if $\epsilon \geq \frac{1}{2} - \frac{1}{2k}$ then reliable computation by formulas is impossible using $\epsilon$-noisy, $k$-input gates. Soon after, Feder [3] extended this result to general circuits, proving reliable computation by circuits is impossible if $\epsilon \geq \frac{1}{2} - \frac{1}{2k}$. Evans and Schulman [4] improved this bound to $\epsilon \geq \frac{1}{2} - \frac{1}{2\sqrt{k}}$.

The above papers developed a certain information-theoretic technique, which yielded both the bounds cited, and lower bounds on noisy circuit depth. However, in 1991, Hajek and Weller used a completely different technique to prove a tight threshold for reliable computation by formulas with noisy 3-input gates [5], showing that $\epsilon < 1/6$ allows reliable computation but $\epsilon \geq 1/6$ forbids it. In this paper, we extend the work of Hajek and Weller to prove a tight threshold for reliable computation by formulas using noisy $k$-input gates ($k$ odd). The main result of this paper is summarized in:

*Theorem 1:* For $k$ odd and

$$\beta_k = \frac{1}{2} - \frac{2^{k-2}}{k\binom{k-1}{\frac{k-1}{2}}}, \quad (2)$$

there exists $\delta < 1/2$ such that all Boolean functions can be $(1-\delta)$-reliably computed by noisy formulas if and only if $\epsilon < \beta_k$. (Using Stirling's approximation, $\beta_k \approx \frac{1}{2} - \frac{\sqrt{\pi}}{2\sqrt{2k}}$ for large values of $k$.) Figure 1 shows how this exact threshold compares to previous bounds for reliable computation.

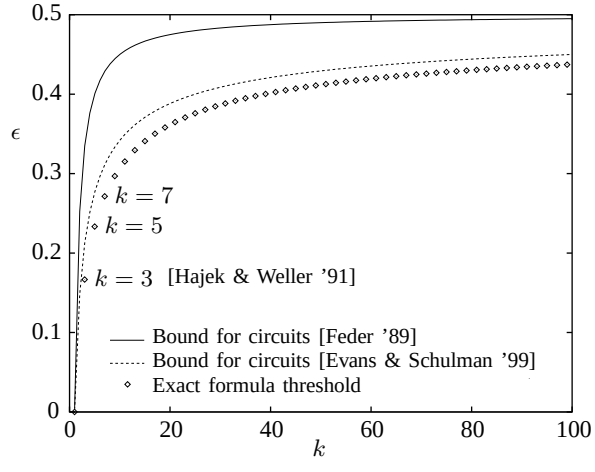


Fig. 1. Bounds for reliable computation.

## II. Threshold Value

To calculate the threshold for reliable computation using $k$-input gates, we start by generalizing von Neumann's expression for the

W. Evans is with the University of British Columbia and is supported in part by the Natural Sciences and Engineering Research Council of Canada. L. Schulman is with Caltech and is supported in part by the National Science Foundation and the Charles Lee Powell Foundation.

[1]The *depth* of a circuit is the number of gates on the longest path from an input of the circuit to its output.

error probability of a 3-input majority gate (1). Let

$$m_{\epsilon,k}(a) = (1-\epsilon)\phi_k(\lfloor k/2 \rfloor, a) + \epsilon(1-\phi_k(\lfloor k/2 \rfloor, a)) \quad (3)$$

where

$$\phi_k(l,a) = \sum_{i=0}^{l} \binom{k}{i} (1-a)^i a^{k-i}.$$

The value $m_{\epsilon,k}(a)$ is the probability that an $\epsilon$-noisy, $k$-input majority gate is incorrect given that its inputs are incorrect independently with probability $a$.

For $k = 3$, if $\epsilon \geq 1/6$ then $m_{\epsilon,3}(a) > a$ for all $a \in [0, 1/2)$. This is von Neumann's observation that, if $\epsilon$ is large, the output of a noisy, 3-input majority gate is less reliable than its inputs. In this section, we generalize von Neumann's observation to noisy, $k$-input gates.

*Lemma 1:* For $k$ odd,

1) if $\epsilon \geq \beta_k$ then $m_{\epsilon,k}(a) > a$ for all $a \in [0, 1/2)$
2) if $\epsilon < \beta_k$ then there exists $\nu_{\epsilon,k} \in [0, 1/2)$ such that $m_{\epsilon,k}(\nu_{\epsilon,k}) = \nu_{\epsilon,k}$ and
   - if $a < \nu_{\epsilon,k}$ then $m_{\epsilon,k}(a) > a$
   - if $a > \nu_{\epsilon,k}$ then $m_{\epsilon,k}(a) < a$

where $\beta_k$ is defined in (2).
See figure 2 for an example of $m_{\epsilon,k}$ when $\epsilon = \beta_k$ and $\epsilon < \beta_k$ (for $k = 3$).

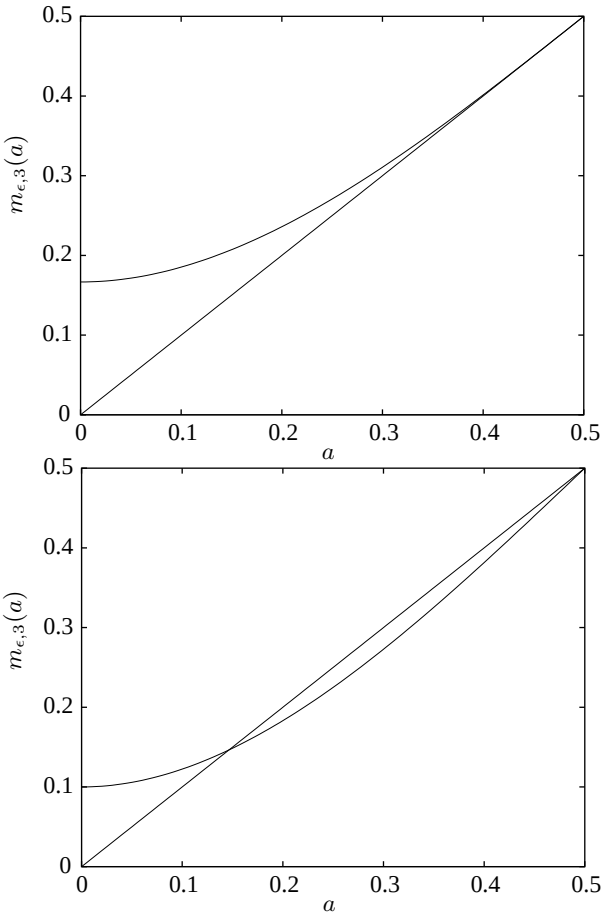


Fig. 2. The error probability of an $\epsilon$-noisy, 3-input majority gate as a function of input error probability for $\epsilon = \beta_3$ and for $\epsilon < \beta_3$.

*Proof:* We first prove that for $k$ odd, $m_{0,k}(a) \leq a$ and $m_{\beta_k,k}(a) > a$ for $a \in [0, 1/2)$. This and the linearity of $m_{\epsilon,k}(a)$ in $\epsilon$ prove the first statement in the lemma.

$m_{0,k}(a)$ is the probability that the noiseless majority of $k$ inputs is incorrect given that each input is incorrect with probability $a$. To show $m_{0,k}(a) \leq a$, we prove the inequality

$$\begin{aligned} m_{0,k}(a) &= \sum_{i=0}^{\lfloor k/2 \rfloor} \binom{k}{i} (1-a)^i a^{k-i} \\ &\leq a \sum_{i=0}^{k} \binom{k}{i} (1-a)^i a^{k-i} = a, \end{aligned}$$

which for $k$ odd is equivalent to

$$\sum_{i=0}^{\lfloor k/2 \rfloor} \binom{k}{i} (1-a)^i a^{k-i} \leq a \sum_{i=0}^{\lfloor k/2 \rfloor} \binom{k}{i} ((1-a)^i a^{k-i} + (1-a)^{k-i} a^i).$$

This inequality holds term by term since if $a = 0$, all terms are zero and otherwise, dividing the $i$th term of the right-hand side by the $i$th term on the left gives,

$$a\left(1 + \left(\frac{1-a}{a}\right)^{k-2i}\right) \geq a\left(1 + \frac{1-a}{a}\right) = 1.$$

To prove $m_{\beta_k,k}(a) > a$ for $a \in [0, 1/2)$, write $a = (1-\alpha)/2$ and let

$$f_k(\epsilon, \alpha) = m_{\epsilon,k}((1-\alpha)/2) - \frac{1-\alpha}{2}$$

(i.e., $f_k(\epsilon, \alpha)$ is the difference between the error probability of the output and the error probability of the inputs). Since for $k$ odd $f_k(\epsilon, 0) = 0$ (in particular, $f_k(\beta_k, 0) = 0$), it suffices for the first statement in the lemma to prove $f_k(\beta_k, \alpha)$ is an increasing function of $\alpha \in (0, 1]$ (i.e. $df_k(\beta_k, \alpha)/d\alpha > 0$),

$$\frac{df_k(\epsilon,\alpha)}{d\alpha} = 1/2 + (1-2\epsilon)\frac{d}{d\alpha}\phi_k\left(\lfloor k/2 \rfloor, \frac{1-\alpha}{2}\right). \quad (4)$$

Since

$$\begin{aligned} \frac{d}{d\alpha}\phi_k\left(\lfloor k/2 \rfloor, \frac{1-\alpha}{2}\right) &= \frac{d}{d\alpha}\sum_{i=0}^{\lfloor k/2 \rfloor}\binom{k}{i}\left(\frac{1+\alpha}{2}\right)^i\left(\frac{1-\alpha}{2}\right)^{k-i} \\ &= -\sum_{i=0}^{\lfloor k/2 \rfloor}\frac{k-i}{2}\binom{k}{i}\left(\frac{1+\alpha}{2}\right)^i\left(\frac{1-\alpha}{2}\right)^{k-i-1} \\ &\quad + \sum_{i=0}^{\lfloor k/2 \rfloor}\frac{i}{2}\binom{k}{i}\left(\frac{1+\alpha}{2}\right)^{i-1}\left(\frac{1-\alpha}{2}\right)^{k-i} \\ &= -\sum_{i=0}^{\lfloor k/2 \rfloor}\frac{k}{2}\binom{k-1}{i}\left(\frac{1+\alpha}{2}\right)^i\left(\frac{1-\alpha}{2}\right)^{k-i-1} \\ &\quad + \sum_{i=0}^{\lfloor k/2 \rfloor - 1}\frac{k}{2}\binom{k-1}{i}\left(\frac{1+\alpha}{2}\right)^i\left(\frac{1-\alpha}{2}\right)^{k-i-1} \\ &= -\frac{k}{2^k}\binom{k-1}{\lfloor k/2 \rfloor}(1-\alpha^2)^{\lfloor k/2 \rfloor}, \end{aligned}$$

substituting into (4) yields,

$$\frac{df_k(\epsilon,\alpha)}{d\alpha} = 1/2 - (1-2\epsilon)\frac{k}{2^k}\binom{k-1}{\frac{k-1}{2}}(1-\alpha^2)^{\frac{k-1}{2}}. \quad (5)$$

Setting $\epsilon = \beta_k$,

$$\frac{df_k(\beta_k,\alpha)}{d\alpha} = \frac{1}{2} - \frac{(1-\alpha^2)^{\frac{k-1}{2}}}{2}$$

which is positive for $\alpha \in (0, 1]$.

We now prove the second statement of the lemma. The second derivative of $f_k(\epsilon, \alpha)$ is

$$\frac{d^2 f_k(\epsilon,\alpha)}{d\alpha^2} = (1-2\epsilon)\frac{k(k-1)}{2^k}\binom{k-1}{\frac{k-1}{2}}\alpha(1-\alpha^2)^{\frac{k-3}{2}}$$

which is nonnegative for all $\alpha \in [0,1]$ and $\epsilon \in [0,1/2]$. Thus $f_k(\epsilon,\alpha)$ is convex in $\alpha \in [0,1]$. Since $f_k(\epsilon,0) = 0$ for $k$ odd and $f_k(\epsilon,1) = \epsilon$, the convexity of $f_k(\epsilon,\alpha)$ will imply the lemma if we can prove $df_k(\epsilon,\alpha)/d\alpha < 0$ at $\alpha = 0$. By equation (5), for $\epsilon < \beta_k$,

$$\frac{df_k(\epsilon,\alpha)}{d\alpha} \leq \frac{1}{2} - \frac{(1-\alpha^2)^{\frac{k-1}{2}}}{2}$$

with equality if and only if $\alpha = 1$, and thus at $\alpha = 0$, $df_k(\epsilon,\alpha)/d\alpha < 0$. ∎

## III. Negative Result

Suppose we simply wish to "remember" an input bit for $L$ computation steps — that is, to design a noisy circuit of depth $L$ with one input $x$ whose output is $x$ with high probability. This is a prerequisite for computing non-trivial functions of many variables. If the computation components are $\epsilon$-noisy, $k$-input gates, the obvious method is to take the majority of $k$ independent copies of the best circuit for remembering the input bit for $L-1$ steps. This construction results in a depth $L$ formula of majority gates that has error probability $m^{(L)}_{\epsilon,k}(0)$ where $m^{(L)}_{\epsilon,k}$ is the $L$-fold composition of $m_{\epsilon,k}$. By lemma 1, if $\epsilon \geq \beta_k$, this technique will not work for arbitrarily large $L$. In fact, for $k$ odd,

$$\text{if } \epsilon \geq \beta_k \text{ then } \lim_{L\to\infty} m^{(L)}_{\epsilon,k}(0) = 1/2.$$

This is the intuitive reason why $\beta_k$ (which is derived from the behavior of noisy, $k$-input majority gates) is the noise threshold for computation using arbitrary noisy $k$-input gates.

To derive a precise statement from this intuition, we prove that if $\epsilon \geq \beta_k$ then, for any fixed $\delta < 1/2$, there are Boolean functions that cannot be computed by formulas with error probability $\delta$. In particular, theorem 2 implies that for sufficiently large $n$, no function that depends[2] on $n$ variables can be computed with error probability $\delta$.

*Theorem 2:* For $k$ odd, if $\epsilon \geq \beta_k$ then any formula using $\epsilon$-noisy, $k$-input gates for computing a Boolean function that depends on at least $k^{L-1}+1$ variables errs with probability $\geq m^{(L)}_{\epsilon,k}(0)$ on some input.

Note that this implies reliable computation is impossible if $\epsilon \geq \beta_k$ (since $\lim_{L\to\infty} m^{(L)}_{\epsilon,k}(0) = 1/2$).

*Proof:* Let $f$ be a Boolean function that depends on at least $k^{L-1}+1$ variables. Let $F$ be a formula for $f$ composed of $\epsilon$-noisy, $k$-input gates. Since $f$ depends on $k^{L-1}+1$ variables, there exists some variable $x$ that is an input only to gates at layers[3] $\geq L$ in $F$. Thus any path from the input $x$ to the output of the formula must pass through at least $L$ gates. Fix the inputs other than $x$ so that either $f = x$ or $f = 1-x$; without loss of generality say $f = x$. Let $F_x$ be the formula $F$ after the inputs other than $x$ have been fixed as above.

Consider the two conditional probabilities $\mathbf{P}[F_x = 1|x = 0]$ and $\mathbf{P}[F_x = 0|x = 1]$. The maximum of these two quantities is a lower bound on the error probability of $F$.

Following Hajek and Weller, one may view these conditional probabilities geometrically as the point $(\mathbf{P}[F_x = 1|x = 0], \mathbf{P}[F_x = 0|x = 1])$ in the unit square. In general, if $Y$ is a Boolean random variable jointly distributed with $x$, let

$$\lambda^Y = (\lambda^Y_0, \lambda^Y_1) = (\mathbf{P}[Y = 1|x = 0], \mathbf{P}[Y = 0|x = 1]).$$

[2]A function depends on an argument $x$ if there exists a setting of the other arguments such that the function restricted to that setting is not a constant.

[3]The *layer* of a gate is the number of gates on the path from its input to the output of the formula.

For example, the $\epsilon$-noisy, $k$-input majority gate with all inputs equal to $x$, produces an output $Y$ described by the point $\lambda^Y = (m_{\epsilon,k}(0), m_{\epsilon,k}(0)) = (\epsilon,\epsilon)$. In this case, the probability that $Y$ differs from $x$ is $\epsilon$.

The gate whose output is $F_x$ (the top gate in the formula) does not receive $x$ directly as input. The value of $x$ must pass through at least $L-1$ noisy gates to reach this top gate. Each gate adds noise to the value of $x$, but the computation performed by the gate may compensate for this noise.

We show that if $\epsilon \geq \beta_k$ then each gate cannot compensate for the added noise. In fact, the space of points $\lambda^Y$, describing possible distributions at the gate's output, contracts as we pass $x$ through more and more noisy gates. In particular, let $S(a)$ be the convex hull of the points $\{(0,1),(1,0),(a,a),(1-a,1-a)\}$. We prove (lemma 2) that if the inputs to an $\epsilon$-noisy, $k$-input gate are described by points in $S(a)$, then the output must lie in $S(m_{\epsilon,k}(a))$. See figure 3.

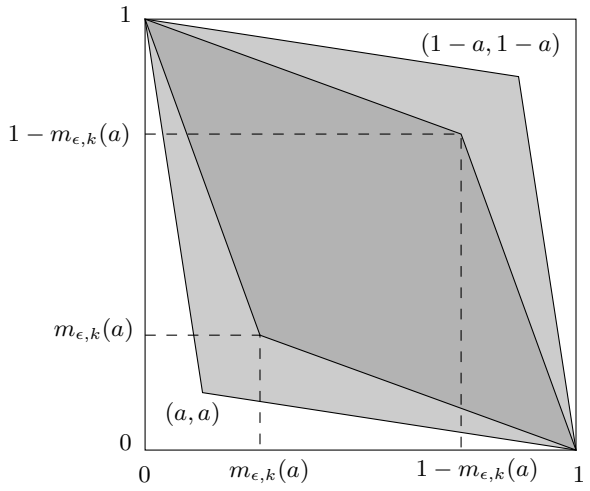


Fig. 3. Contraction of $S(a)$ (light gray) to $S(m_{\epsilon,k}(a))$ (dark gray) caused by one noisy gate.

Using this lemma, we prove by induction on $L$ that the point describing the output of $F_x$ lies within $S(m^{(L)}_{\epsilon,k}(0))$. This establishes the theorem since any random variable $Y$ whose point lies in $S(a)$ differs from $x$ with probability at least $a$. Thus, the error probability of $F_x$ is at least $m^{(L)}_{\epsilon,k}(0)$.

For $L = 1$, the formula consists of at least one gate. The points describing inputs to the top gate of the formula $F_x$ lie within $S(0)$ (trivially) and thus, by lemma 2, the point describing the output lies within $S(m_{\epsilon,k}(0))$.

For $L > 1$, the formula consists of a top gate with at most $k$ inputs. Each of these inputs is either constant with respect to $x$ or the output of a formula in which $x$ is an input to gates at layers $\geq L-1$. In the first case, the point describing the input lies within $S(a)$ for all $a$. In the second, the point describing the input lies in $S(m^{(L-1)}_{\epsilon,k}(0))$ by induction. Thus, by lemma 2, the point describing the output lies within $S(m^{(L)}_{\epsilon,k}(0))$. ∎

## IV. Contraction of $S(a)$

*Lemma 2:* If $\epsilon \geq \beta_k$ and $\lambda^{Y_1}, \lambda^{Y_2}, ..., \lambda^{Y_k} \in S(a)$ with $a \in [0,1/2]$ then for all $\epsilon$-noisy, $k$-input gates $g$ with inputs $Y_1, Y_2, ..., Y_k$ and output $Y$, $\lambda^Y \in S(m_{\epsilon,k}(a))$.

*Proof:* We will prove in lemmas 3 and 4 that we may assume that $\lambda^{Y_i} = (a,a)$ for all $i$ and that $g$ is an $\epsilon$-noisy, $k$-input threshold gate. A $k$-input threshold gate outputs 1 if and only if the number of inputs equal to 1 is at least $t$. The threshold $t$ is an integer between 0 and $k$ inclusive.

We prove that the output $Y$ of the gate $g$ has $\lambda^Y \in S(m_{\epsilon,k}(a))$. By symmetry, we need only consider those threshold gates $g$ with threshold $t \geq \lceil k/2 \rceil$. We will prove that $\lambda^Y$ lies within the convex hull of the vertices $\{(0,1),(1/2,1/2),(m_{\epsilon,k}(a),m_{\epsilon,k}(a))\}$. Since $t \geq \lceil k/2 \rceil$, $\lambda_0^Y \leq \lambda_1^Y$. Also, $a \leq 1-a$ implies $\lambda_0^Y + \lambda_1^Y \leq 1$. Thus we need only prove that,

$$\lambda_0^Y + m_{\epsilon,k}(a)(\lambda_1^Y - \lambda_0^Y) \geq m_{\epsilon,k}(a) \tag{6}$$

when $\epsilon \geq \beta_k$.

Let $V$ be the pre-noise[4] output of gate $g$. That is, $\lambda_b^Y = \epsilon + (1-2\epsilon)\lambda_b^V$ for $b \in \{0,1\}$. Then (6) becomes,

$$\lambda_0^V + m_{\epsilon,k}(a)(\lambda_1^V - \lambda_0^V) \geq \phi_k(\lfloor k/2 \rfloor, a).$$

Since $g$ is a $k$-input, threshold $t$ gate, $\lambda_0^V = \phi_k(k-t,a)$ and $\lambda_1^V = \phi_k(t-1,a)$. Since $\epsilon \geq \beta_k$ implies $m_{\epsilon,k}(a) > a$, it suffices to prove $\lambda_0^V + a(\lambda_1^V - \lambda_0^V) \geq \phi_k(\lfloor k/2 \rfloor, a)$ or,

$$a(\lambda_1^V - \phi_k(\lfloor k/2 \rfloor, a)) \geq (1-a)(\phi_k(\lfloor k/2 \rfloor, a) - \lambda_0^V)$$

Substituting the values of $\phi_k(\lfloor k/2 \rfloor, a)$, $\lambda_0^V$, and $\lambda_1^V$ yields,[5]

$$a(\phi_k(t-1,a) - \phi_k(\lfloor k/2 \rfloor, a)) \geq (1-a)(\phi_k(\lfloor k/2 \rfloor, a) - \phi_k(k-t,a))$$

After expanding each side as a summation, the inequality holds term by term since $a \in [0,1/2]$ and $i \geq \lceil k/2 \rceil$ imply $a\binom{k}{i}(1-a)^i a^{k-i} \geq (1-a)\binom{k}{k-i}(1-a)^{k-i}a^i$. ∎

## V. Reduction lemmas

The above proof relies on two lemmas that are rather straightforward extensions of similar lemmas for $k = 3$ given by Hajek and Weller [5].

An $\epsilon$-noisy, $k$-input gate $g$ takes as input $Y_1, \ldots, Y_k$, described by points $\lambda^{Y_1}, \ldots, \lambda^{Y_k}$, and outputs $Y$ described by $\lambda^Y$. Thus the gate $g$ defines a mapping $g : [0,1]^{2k} \to [0,1]^2$. Lemma 2 states that if $\epsilon \geq \beta_k$ then the union over all $g$ of $g(S(a)^k)$ is contained in $S(m_{\epsilon,k}(a))$. The purpose of the following two lemmas is to show that it suffices to prove that the union over all threshold gates $g$ of $g((a,a)^k)$ is contained in $S(m_{\epsilon,k}(a))$. (Note: $(a,a)^k$ is the point $(a,a),(a,a),...,(a,a)$ in $[0,1]^{2k}$.) The method is to show that the set of image points has the same convex hull in both cases. Thus, since $S(m_{\epsilon,k}(a))$ is convex, showing containment of either set implies containment of the other.

*Lemma 3:* If $C$ is the convex hull of the union over all $g$ of $g(S(a)^k)$ and $C_a$ is the convex hull of the union over all $g$ of $g((a,a)^k)$ then

$$C = C_a$$

*Proof:* The mapping from $S(a)^k$ to $[0,1]^2$ defined by $g$ is affine, $[0,1]^2 \to [0,1]^2$, in each $\lambda^{Y_i}$ when the others are fixed. Thus the image of $S(a)^k$ is contained in the convex hull of the image of the set of vertices of $S(a)^k$. Each vertex is of the form $(\lambda^{Y_1}, \lambda^{Y_2}, ..., \lambda^{Y_k})$ with $\lambda^{Y_i} \in \{(1,0),(0,1),(a,a),(1-a,1-a)\}$. If $\lambda^{Y_i} \in \{(1,0),(0,1)\}$ then the same value of $\lambda^Y$ can be obtained with $\lambda^{Y_i} = (a,a)$ by modifying the gate $g$ to ignore the value of $Y_i$. Similarly, if $\lambda^{Y_i} = (1-a,1-a)$ then the same value of $\lambda^Y$ can be obtained with $\lambda^{Y_i} = (a,a)$ by modifying the gate $g$ to negate input $Y_i$. The lemma follows. ∎

[4]An $\epsilon$-noisy gate computes a Boolean function of its inputs that is then complemented with probability $\epsilon$ to become the gate's output. The value computed by the gate prior to the probabilistic change is the gate's *pre-noise* output.

[5]If $t = \lceil k/2 \rceil$ both sides of the inequality are zero.

*Lemma 4:* If $C_a$ is the convex hull of the union over all $g$ of $g((a,a)^k)$ and $C_{a,t}$ is the convex hull of the union over threshold gates $g$ of $g((a,a)^k)$ then

$$C_a = C_{a,t}$$

*Proof:* Note that $\lambda^{Y_i} = (a,a)$ for all $i$. To establish the lemma, it suffices to prove that for any constants $r$ and $s$, $r\lambda_0^Y + s\lambda_1^Y$ is minimized when $g$ is some threshold function.

Again let $V$ be the pre-noise output of gate $g$, so $\lambda_b^Y = \epsilon + (1-2\epsilon)\lambda_b^V$ for $b \in \{0,1\}$. Thus to minimize $r\lambda_0^Y + s\lambda_1^Y$, we minimize $r\lambda_0^V + s\lambda_1^V$.

$$\lambda_0^V = \sum_{(Y_1,Y_2,...,Y_k)\in S_1} \mathbf{P}[(Y_1,Y_2,...,Y_k)|x=0]$$
$$\lambda_1^V = \sum_{(Y_1,Y_2,...,Y_k)\in S_0} \mathbf{P}[(Y_1,Y_2,...,Y_k)|x=1]$$

where $S_b$ is the set of $k$-bit vectors representing inputs for which $V = b$. A gate $g$ that minimizes $r\lambda_0^V + s\lambda_1^V$ has $(Y_1,Y_2,...,Y_k) \in S_1$ if and only if $r\mathbf{P}[(Y_1,Y_2,...,Y_k)|x=0] < s\mathbf{P}[(Y_1,Y_2,...,Y_k)|x=1]$. From the fact that $\lambda^{Y_i} = (a,a)$,

$$\mathbf{P}[(Y_1,Y_2,...,Y_k)|x=0] = a^t(1-a)^{k-t}$$
$$\mathbf{P}[(Y_1,Y_2,...,Y_k)|x=1] = a^{k-t}(1-a)^t$$

where $t$ is the number of ones in the vector $(Y_1,Y_2,...,Y_k)$. Thus the relation $r\mathbf{P}[(Y_1,Y_2,\ldots,Y_k)|x=0] < s\mathbf{P}[(Y_1,Y_2,\ldots,Y_k)|x=1]$ holds monotonically in $t$ and the lemma follows. ∎

## VI. Positive Result

The preceding section shows that the ability of an $\epsilon$-noisy, $k$-input majority gate to decrease error probability is necessary for reliable computation using $k$-input gates. (Put another way, reliable computation is not possible unless a bit can be maintained indefinitely in a formula by repeatedly taking majority.) In this section, we show that this is also a sufficient condition.

For $\epsilon < \beta_k$, we prove that there exists $\delta < 1/2$ such that given any Boolean function, we can construct a formula using $\epsilon$-noisy, $k$-input gates that $(1-\delta)$-reliably computes the function. One obvious idea is to use von Neumann's technique of taking the noisy majority of $k$ independent copies of a computation in order to decrease the error probability. This process can be repeated to decrease the error probability still further, but there is a limit. It will decrease the error probability if and only if the original error probability is in the interval $(\nu_{\epsilon,k}, 1/2)$ where

$$\nu_{\epsilon,k} = \lim_{L\to\infty} m_{\epsilon,k}^{(L)}(a)$$

(By lemma 1, the limit exists and is the same for any $a \in [0,1/2)$.)

Once the error probability is back to a reasonable level, more computation can be done. Such a scheme works as long as computation can be performed at the "reasonable" error probability level achieved by the majority gates. In other words, computation at this level of error probability must result in an output that is correct on all inputs with probability strictly greater than $1/2$.

Hajek and Weller found a noisy, 3-input gate that computes reliably given very noisy inputs. Strangely at first sight, it requires the error probability of all of its inputs to be close to $\nu_{\epsilon,k}$, not just close to or less than $\nu_{\epsilon,k}$. In fact, the probability of an incorrect output bit can increase from below 1/2 to above 1/2 by *decreasing* the error probability of some of the inputs. Thus, if we know only that the noise at each gate is at most $\epsilon$, an adversary could decrease the noise of some gates to below $\epsilon$ and ruin the reliability of the output. The construction takes advantage of the precise $\epsilon$ noise at the gates to obtain a reliable formula.

Hajek and Weller's noisy, 3-input computation gate is used to simulate the computation of a noiseless 2-input NAND gate. It is called an XNAND gate. A noiseless XNAND gate outputs 1 for inputs (0,0,0), (1,0,0), (0,0,1), and (0,1,1); and outputs 0 otherwise.

Let $x$ and $y$ be the inputs to a NAND gate. Let $X$ be a noisy version of $x$; and $Y_1$ and $Y_2$ independent noisy versions of $y$. The output of XNAND on input $(X, Y_1, Y_2)$ is intended to be a reliable version of NAND on input $(x, y)$. The following lemma makes this connection precise.

*Lemma 5 (Lemma 3.1 [5]):* For $\epsilon, \nu \in [0, 1/2)$ there is a $\delta < 1/2$ and an open interval $I$ with $\nu \in I \subset [0, 1/2]$ so that the following is true. If $\mathbf{P}[X \neq x]$, $\mathbf{P}[Y_1 \neq y]$, $\mathbf{P}[Y_2 \neq y] \in I$, and if $Z$ is the output of an $\epsilon$-noisy XNAND gate with input $(X, Y_1, Y_2)$, then $\mathbf{P}[Z \neq \text{NAND } (x, y)] < \delta$.

*Proof:* If $\mathbf{P}[X \neq x] = \mathbf{P}[Y_1 \neq y] = \mathbf{P}[Y_2 \neq y] = \nu$ then $\mathbf{P}[Z \neq \text{NAND } (x, y)]$ equals $(1 - \nu)(2\epsilon - 1) + 1 - \epsilon$ if $(x, y) = (0, 0)$ or $(1, 1)$ and equals $(2\nu^2 - 2\nu + 1)(2\epsilon - 1) + 1 - \epsilon$ if $(x, y) = (1, 0)$ or $(0, 1)$. In either case, if $\epsilon, \nu \in [0, 1/2)$ then $\mathbf{P}[Z \neq \text{NAND } (x, y)] < 1/2$. Since $\mathbf{P}[Z \neq \text{NAND } (x, y)]$ is a continuous function of $(\mathbf{P}[X \neq x], \mathbf{P}[Y_1 \neq y], \mathbf{P}[Y_2 \neq y])$, the proof is complete. ∎

We use the XNAND gate in conjunction with $k$-input majority gates ($k$ odd) to prove that reliable computation by precisely $\epsilon$-noisy, $k$-input gates is possible if $\epsilon < \beta_k$.

*Theorem 3:* For $k$ odd and $0 \leq \epsilon < \beta_k$, there exists $\delta < 1/2$ such that any Boolean function can be $(1 - \delta)$-reliably computed by a formula using $\epsilon$-noisy, $k$-input gates.

*Proof:* The proof is a simple extension of Proposition 3 from Hajek and Weller [5]. For $k$ odd ($k \geq 3$), an $\epsilon$-noisy XNAND gate can be implemented by an $\epsilon$-noisy, $k$-input gate that ignores all but three of its inputs. Use $\delta$ and $I$ with $\nu = \nu_{\epsilon,k}$ from the proof of lemma 5 and choose $L$ large enough so that $[m_{\epsilon,k}^{(L)}(0), m_{\epsilon,k}^{(L)}(\delta)] \subset I$.

Start with a formula composed of 2-input NAND gates that computes the function. The idea is to replace the noiseless formula with a formula composed of $\epsilon$-noisy, $k$-input majority gates and XNAND gates. The replacement is performed inductively. If the formula is trivial, i.e. a single input or constant, then we are done. Otherwise, suppose the top NAND gate has two inputs $x$ and $y$. By induction, replace the formulas computing $x$ and $y$ with three noisy formulas: one that computes a noisy version $U$ of $x$, and two that compute independent noisy versions, $V_1$ and $V_2$, of $y$. The induction insures that the error probabilities of these noisy versions lie within $[0, \delta]$.

By replicating their formulas, make $k^L$ independent copies of each of $U$, $V_1$,and $V_2$. Use $L$ levels of $\epsilon$-noisy, $k$-input gates to combine the copies of $U$ into one noisy version $X$ of $x$ whose error probability lies within $I$. Do the same with the copies of $V_1$ and $V_2$ to obtain $Y_1$ and $Y_2$ with error probability in $I$. By lemma 5, the output of an XNAND gate with these inputs will be a $(1 - \delta)$-reliable version of the original output. ∎

## VII. Conclusions

This paper extends the work of Hajek and Weller [5] to establish an exact threshold for reliable computation by formulas using $\epsilon$-noisy, $k$-input gates for odd $k$. Since a $k+1$-input gate can simulate a $k$-input gate, our results for odd $k$ translate into bounds on noise levels that permit reliable computation for even $k$. However, the value (or even the existence) of a threshold for reliable computation by formulas using $k$-input gates for even $k$ is unknown. Evans and Pippenger [6] made some progress in this direction, showing that if a formula is constructed from independent $\epsilon$-noisy, 2-input NAND gates then reliable computation can or cannot take place depending on whether $\epsilon$ is less than or greater than $(3 - \sqrt{7})/4 = 0.08856\ldots$. In addition, the existence and value of a threshold for reliable computation by circuits using $\epsilon$-noisy, $k$-input gates for even or odd $k$ is unknown.

## Acknowledgements

We would like to thank the anonymous referees for their comments.

## References

[1] J. von Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," in *Automata Studies*, C. E. Shannon and J. McCarthy, Eds. Princeton University Press, 1956, pp. 43–98.

[2] N. Pippenger, "Reliable computation by formulas in the presence of noise," *IEEE Trans. Inform. Theory*, vol. 34, no. 2, pp. 194–197, Mar. 1988.

[3] T. Feder, "Reliable computation by networks in the presence of noise," *IEEE Trans. Inform. Theory*, vol. 35, no. 3, pp. 569–571, May 1989.

[4] W. Evans and L. J. Schulman, "Signal propagation and noisy circuits," *IEEE Trans. Inform. Theory*, vol. 45, no. 7, pp. 2367–2373, 1999.

[5] B. Hajek and T. Weller, "On the maximum tolerable noise for reliable computation by formulas," *IEEE Trans. Inform. Theory*, vol. 37, no. 2, pp. 388–391, Mar. 1991.

[6] W. Evans and N. Pippenger, "On the maximum tolerable noise for reliable computation by formulas," *IEEE Trans. Inform. Theory*, vol. 44, no. 3, pp. 1299–1305, May 1998.

**William S. Evans** William S. Evans received the B.Sc.degree from Yale University in 1987 and the Ph.D. degree from the University of California at Berkeley in 1994 both in computer science. In 1994, he joined the Department of Computer Science at the University of Arizona as an assistant professor. Since 2001, he has been an assistant professor in the Department of Computer Science at the University of British Columbia. His research interests include information theory, algorithms, computational geometry, and compression.

**Leonard J. Schulman** Leonard J. Schulman received his B.Sc. in Mathematics in 1988 and his Ph.D. in Applied Mathematics in 1992, both from the Massachusetts Institute of Technology. He has held appointments as NSF mathematical sciences postdoctoral fellow at UC Berkeley, visiting scientist at the Weizmann Institute of Science, assistant and associate professor at the Georgia Institute of Technology, and visiting member at the Mathematical Sciences Research Institute. He is a recipient of the Jon A. Bucsela prize in mathematics at MIT and an NSF CAREER award. Since 2000 he has been associate professor of computer science at the California Institute of Technology.