The number of Boolean functions computed by formulas of a given size.

Petr Savický* Alan R. Woods

Abstract

Estimates are given of the number $B(n, L)$ of distinct functions computed by propositional formulas of size L in n variables, constructed using only literals and $\wedge, \vee$ connectives. (L is the number of occurrences of variables. $L-1$ is the number of binary $\wedge$'s and $\vee$'s. $B(n, L)$ is also the number of functions computed by two terminal series-parallel networks with L switches.) Enroute the read-once functions, which are closely related to Schröder numbers, are enumerated.

Writing $B(n, L) = b(n, L)^L$, we find that if L and $\beta(n)$ go to infinity with increasing n and $L \leq 2^n/n^{\beta(n)}$, then $b(n, L) \sim cn$, where $c = 2/(\ln 4 - 1)$. Making a comparison with polynomial size Boolean circuits, this implies the following. For any constant $\alpha > 1$, almost all Boolean functions with formula complexity at most n^α, cannot be computed by any circuit constructed from literals and fewer than $\alpha^{-1} n^\alpha$ two input $\wedge, \vee$ gates.

1 Introduction.

A very early discovery in the asymptotic study of propositional formulas, was the theorem of Riordan and Shannon [23] that for almost all Boolean functions $f(X_1, \ldots, X_n)$ of n propositional variables $X_1, \ldots, X_n$, the minimum formula complexity $\mathbf{L}(f)$ is exponential, satisfying

$$\mathbf{L}(f) \;>\; \frac{2^n}{\log n}\left(1 - \frac{\kappa}{\log n}\right) \tag{1}$$

*The first author was partially supported by EU Grant no. INCO-COP 96-0195 "ALTEC-KIT".

for some constant κ. This was complemented by an ingenious construction of Lupanov [12, 13], who established that for any Boolean function $f(X_1, \ldots, X_n)$,

$$\mathbf{L}(f) \;\leq\; \frac{2^n}{\log n}\left(1 + \mathrm{O}\left(\frac{\log\log n}{\log n}\right)\right). \tag{2}$$

Put together, these show

$$\mathbf{L}_{\max}(n) \;\sim\; \frac{2^n}{\log n}, \quad \text{where} \;\; \mathbf{L}_{\max}(n) = \max_f \mathbf{L}(f). \tag{3}$$

Surprisingly, given the interest in polynomial size formulas as a model for feasible parallel computation, and the importance of determining the relative computational power of polynomial size formulas versus polynomial size circuits, little attention seems to have been applied to estimating the number $B(n, L)$ of Boolean functions which can be computed by formulas of an arbitrary size L. This problem generalizes that of approximating $\mathbf{L}_{\max}$, since $\mathbf{L}_{\max}$ is the smallest solution L of the equation $B(n, L) = 2^{2^n}$. However the main emphasis in the present paper, will be placed on estimating $B(n, L)$ for L of moderate magnitude. The problem is of particular interest when $L = n^\alpha$ with $\alpha \geq 1$ constant, but we present bounds for all $L \leq \mathbf{L}_{\max}(n)$.

Probabilistic methods appear to be very powerful in combinatorics and computer science. A natural point of view, is that a probabilistic method involves the investigation of the properties of a typical object chosen from some (usually finite) set. In particular, the lower bound (1) was proved, in fact, as a bound on the complexity of a typical Boolean function chosen from the set of all functions of n variables. However, this shows that investigating the typical Boolean function of this kind does not help us understand the properties of functions of moderate complexity. The present paper is a step towards developing techniques for investigating the properties of a typical Boolean function of a given complexity. Even for other models of computation, not a lot is known about function counting problems of this sort. Since formulas are essentially trees, and there are well developed methods for enumerating trees, they seem to be a very appropriate place to start to answer this type of question. Other approaches in related directions may be found in [25], [11], [7].

For our purposes, a *propositional formula* in variables $X_1, \ldots, X_n$ is a rooted tree, with all edges directed toward the root (or *output*), a *literal* X_i

or $\overline{X}_i$ (NOT X_i) attached to each leaf, and a *connective* (or *gate*) $\wedge$ (AND) or $\vee$ (OR) attached to each nonleaf node. Each such formula computes some Boolean function $f(X_1, \ldots, X_n)$ in the obvious way.

The *size* L of the formula, is the number of occurrences of literals in it (i.e., the number of leaves). In the special, but very familiar, case of formulas such as

$$((X_1 \wedge ((X_2 \wedge \overline{X}_3) \vee (\overline{X}_2 \wedge X_3))) \vee ((\overline{X}_1 \wedge X_2) \wedge X_3)) \;, \tag{4}$$

in which all the $\wedge, \vee$ connectives are binary, $L-1$ is also the number of connectives. So, when each $\wedge, \vee$ gate has *fan-in* (in-degree) exactly 2, our asymptotic estimates apply equally well to the number $L-1$ of gates, and can easily be converted into expressions based on the number *wires* (edges), which is $2L-2$.

The *formula complexity* $\mathbf{L}(f)$ of a Boolean function f is the smallest L such that some formula of size L computes f. Notice that this formula can, if we wish, be assumed to have only binary connectives.

Throughout the paper, n will always denote a *sufficiently large* positive integer. $\log n$ denotes the logarithm of n to base 2, while $\ln n$ denotes the natural logarithm.

Let $b(n, L)$ be the "growth rate" of $B(n, L)$ considered as an exponential in L. That is, let $B(n, L) = b(n, L)^L$. It transpires that $b(n, L)$ depends only very weakly on L.

THEOREM 1.1 *Suppose that as $n \to \infty$, $l_1(n) \to \infty$ and $l_2(n)/n^\kappa \to \infty$ for every constant κ. Then uniformly for all L satisfying $l_1(n) \leq L \leq 2^n/l_2(n)$,*

$$b(n, L) \sim cn,$$

or equivalently,

$$\log B(n, L) \;=\; L \log n + L\,(\log c - o(1)),$$

where $c = 2/(\ln 4 - 1)$.

We will use c to denote this constant $c = 2/(\ln 4 - 1) = 5.17739\ldots$ throughout the paper.

For L too large to satisfy the condition of the theorem, we have not been able to prove matching upper and lower bounds. However, the upper bound

$b(n,L) < cn$ still holds, and if $L \le 2^n/n^\epsilon$ for some constant ϵ, then we are able to show that $b(n,L) > \delta n$, where the constant $\delta > 0$ depends on ϵ. For large L, the behaviour of $b(n,L)$ turns out to be intimately related to the error term in (3). An enhancement of Lupanov's construction yields:

THEOREM 1.2 *If $n \le L \le \mathbf{L}_{\max}(n)$, then for n sufficiently large,*

$$\frac{n}{4 \log\log n} \; < \; b(n,L) \; < \; cn.$$

Obviously, $b(n, \mathbf{L}_{\max})^{\mathbf{L}_{\max}} = B(n, \mathbf{L}_{\max}) = 2^{2^n}$. Solving for $\mathbf{L}_{\max}$ reveals that this theorem contains the following improvement of (2):

COROLLARY 1.3

$$\mathbf{L}_{\max}(n) \; \le \; \frac{2^n}{\log n}\left(1 + \frac{\log\log\log n}{\log n} + \mathrm{O}\left(\frac{1}{\log n}\right)\right).$$

These theorems also imply a generalization of (3), which asserts the following fact in the special case $L = \mathbf{L}_{\max}(n)$.

COROLLARY 1.4 *If $L \le \mathbf{L}_{\max}(n)$ and $L \to \infty$ with n, then*

$$\log B(n,L) \; \sim \; L \log n \; .$$

$B(n,L)$ is also the number of functions computed by two terminal *series-parallel networks* of L switches, each of which has its "on" state governed by one of the literals from $X_1, \overline{X}_1, \ldots X_n, \overline{X}_n$. Strictly speaking the paper of Riordan and Shannon was about such networks. It does not mention formulas at all. After that paper, the literature seems to have diverged into two streams — the complexity literature (which is now usually in terms of formulas) and the enumerative literature (which often uses series-parallel networks). For the present paper it is necessary to reunify these streams. More details of the connections can be found in Sections 4 and 6.

Our interest in estimating $B(n,L)$ arose from the question of the relative computational power of polynomial size formulas and polynomial size circuits. *Circuits* are defined similarly to formulas except that rather than being a tree, the underlying structure is a directed graph having no directed cycles. This directed graph is also required to have a unique node with out-degree 0, the *output*, and the property that every edge forms part of some directed path to the output. To each leaf (i.e., node with in-degree 0) is attached a literal, and to each nonleaf, an $\wedge$ or $\vee$ gate is attached. The circuit computes some Boolean function, in the obvious way. The *size* C of a circuit is the number of gates. It is well known, that the size of any circuit with unbounded fan-in gates, is polynomially related to that of a fan-in 2 circuit computing the same function. So, as is traditional, we will restrict our attention to circuits in which every gate has fan-in (in-degree) 2. The *circuit complexity* $\mathbf{C}(f)$ of a Boolean function f is the smallest C such that some circuit of size C, having only literals and $\wedge, \vee$ gates with fan-in 2, computes f. Formulas are circuits in which each gate other than the output has *fan-out* (out-degree) exactly one. Any formula with *formula size* L, is equivalent to a formula having only fan-in 2 gates, the same number L of occurrences of literals, and therefore exactly $L-1$ gates, that is, *circuit size* $L-1$. So

$$\mathbf{C}(f) \leq \mathbf{L}(f).$$

But this raises the question: Is a partial converse true? Given a constant $\alpha \geq 1$ is there always some constant β, such that for any Boolean function $f(X_1, \ldots, X_n)$, if $\mathbf{C}(f) \leq n^{\alpha}$ then $\mathbf{L}(f) \leq n^{\beta}$? This is the important open problem of whether polynomial size formulas have the same computational power as polynomial size circuits.

Polynomial size circuits are rather powerful computationally. For example it is well known that for every decision problem in the polynomial time class P, there is some α such that for each n the corresponding Boolean function $f(X_1, \ldots, X_n)$ has $\mathbf{C}(f) \leq n^{\alpha}$. On the other hand, there are many problems in P which are not known to have polynomial size formulas, even though the parallel time complexity class "nonuniform NC^1" associated with such formulas, certainly possesses nontrivial computational ability. Perhaps just as significantly, the best hope for proving NP $\neq$ P is generally thought to be to exhibit a problem in NP (the nondeterministic polynomial time class) which does not have polynomial size circuits. However formulas are easier to analyse than circuits, so there would appear to be more hope of showing that some problem in NP does not have polynomial size formulas, although even this is open.

Although not proved, it is widely conjectured that there is a sequence functions f_n in n variables, such that $\mathbf{L}(f_n)$ grows exponentially, even though

$\mathbf{C}(f_n)$ is polynomially bounded. Since both the complexity measures in question are nonuniform, for proving their separation, it is not necessary to exhibit an explicit example. An existence proof would be sufficient.

Given this expectation that polynomial size circuits are computationally stronger than polynomial size formulas, a somewhat surprising consequence of even weak versions of our main result is that a random function f, with $\mathbf{L}(f)$ bounded by a polynomial, does not separate circuit size and formula size by more than a constant factor. In other words, for the range of functions f with a polynomially bounded size complexity $\mathbf{L}(f)$, changing from using a formula to using a circuit cannot, on average, decrease the size by more than a constant factor.

THEOREM 1.5 *For any constant $\alpha > 1$, as $n \to \infty$, almost all Boolean functions $f(X_1, \ldots, X_n)$ satisfying $\mathbf{L}(f) \leq n^\alpha$, have*

$$\mathbf{C}(f) \; > \; \alpha^{-1} n^\alpha \; \geq \; \alpha^{-1} \mathbf{L}(f).$$

2 Formulas versus circuits.

Sharpening a standard technique [28, 24, 29, 4], gives the following upper bound, which together with Theorem 1.1, is all that is necessary to deduce Theorem 1.5.

LEMMA 2.1 *For $C \geq 1, n \geq 1$, the number of Boolean functions $f(X_1, \ldots, X_n)$ with $\mathbf{C}(f) \leq C$, is less than $(C + 2n)^{2C}/C! < C^C e^{C+4n}$.*

PROOF: Clearly we may suppose $n \geq 2$, $C \geq 2$. Let $g_1, g_2, \ldots, g_C$ be names of the gates in a circuit. In order to describe a concrete circuit, it is sufficient to attach to each gate one of the connectives $\{\vee, \wedge\}$ and an unordered pair of names of two other gates or literals. There are exactly

$$\left(2\binom{C-1+2n}{2}\right)^C \leq (C+2n)^{2C}$$

such descriptions. Clearly, some of these descriptions do not represent a circuit satisfying all requirements but every correct circuit may be described in this way. Note that the output does not have a special name. In a correct circuit, it is determined by the fact that it is the only gate not used in any other gate.

It is easy to see, although somewhat technical to prove (cf., Lemma 5.1 below), that every function representable by a circuit of size at most C is also representable by a circuit of size exactly C satisfying the additional requirement that no two of its gates compute the same function. It is also easy to see that in a circuit satisfying the last mentioned property, each of the $C!$ permutations of the names of the gates leads to a different description of a circuit computing the same function. So the number of functions f with $\mathbf{C}(f) \le C$ is at most

$$\frac{(C+2n)^{2C}}{C!} \;<\; \frac{(C+2n)^{2C}}{C^C/e^C} \;=\; C^C e^C \left(1+\frac{2n}{C}\right)^{2C} \;<\; C^C e^C \,\left(e^{2n/C}\right)^{2C}.$$

■

Proof of Theorem 1.5: Assuming Theorem 1.1, for any small constant $\epsilon > 0$, the number $B(L,n)$ of functions f with $\mathbf{L}(f) \le L$, satisfies

$$B(n,L) \;>\; (c-\epsilon)^L n^L\,,$$

when n is sufficiently large. By Lemma 2.1, the number of such functions which can be computed by circuits of size at most C, is less than $C^C e^{C+4n} = \mu\, B(n,L)$, say. So the fraction μ satisfies

$$\ln\mu \;<\; C\ln C + C + 4n - L\ln n - L\ln(c-\epsilon)\,.$$

For $L=[n^\alpha]$, $C \le \alpha^{-1} n^\alpha$, $\alpha > 1$, the right hand side is bounded above by a quantity asymptotic to

$$n^\alpha \ln n - \alpha^{-1} n^\alpha \ln\alpha \;+\; \alpha^{-1} n^\alpha \;-\; n^\alpha \ln n - n^\alpha \ln(c-\epsilon)\,,$$

so for all large n,

$$\ln\mu \;<\; n^\alpha - n^\alpha\ln(c-\epsilon) \;=\; -n^\alpha \ln\left(\frac{c-\epsilon}{e}\right)\,.$$

But as $c = 2/(\ln 4 - 1) > e$, we can certainly choose ϵ so that $\ln\mu \to -\infty$, and therefore $\mu \to 0$, as $n \to \infty$.

■

To prove that for any constant $\beta < \alpha^{-1}$, almost all functions f with $\mathbf{L}(f) \geq n^\alpha$ have $\mathbf{C}(f) > \beta n^\alpha$, is technically easier, requiring only the weaker result $\log B(n, L) \sim L \log n$ (Corollary 1.4). Since some readers may be primarily interested in this asymptotic formula, and as much of the effort in this paper is focused on producing arguments which are sharp enough to also give the error term, let us provide some directions for producing a relatively short proof of Corollary 1.4. In this weak form, the upper bound is immediate from a result of Riordan and Shannon [23]. Similar arguments can also be found in standard monographs such as those of Savage [24], Wegener [29], and Dunne [4]. For the lower bound, distinguish two cases, which together cover all possibilities, with some overlap. For $L = 2^{o(n)}$ (or even $L \leq (2-\epsilon)^n$, $\epsilon > 0$ constant), Theorem 7.2 can be used, but only the bound $r_n \geq n!$ is required. This bound is immediate from considering the $2\,n!$ read-once formulas obtained by permuting the variables in

$$X_1 \wedge (X_2 \vee (X_3 \wedge (X_4 \vee \quad \cdots \quad))),$$
$$X_1 \vee (X_2 \wedge (X_3 \vee (X_4 \wedge \quad \cdots \quad))).$$

These occur in pairs, obtained by swapping the innermost pair of variables, but otherwise compute distinct functions by Lemma 4.1. Apart from that lemma, the material in Sections 3, 4 and 5 can be avoided. (In particular there is no need to resort to complex analysis.) If m can be chosen such that $(\log m)/\log n \to 1$ and $L/2^m \to \infty$, which is certainly the case if (say) $L > 2^{n/\log n}$, the result follows by applying Lemma 9.4 (with $g(n) = h(n) = \log n$) to Lupanov's Theorem (our (2) and Corollary 9.2), restated in the weaker form $\log b(n, \mathbf{L}_{\max}) \geq \log n - o(\log n)$.

3 Asymptotic enumeration of trees.

As emphasised in the introduction, formulas are a particular sort of tree. We will have cause to enumerate particular sorts of formulas, by means of a standard method for the asymptotic enumeration of trees, which was expounded by Polya [20, 21], and is based on ideas in complex analysis that came even earlier. In our case, this is applied to generating series equations closely related, or identical, to some appearing in the 19th Century work of Cayley [2], Schröder [26], and MacMahon [14]. Most, but not all, of what we need already appears *somewhere* in the literature, although we did not become cognizant of the full extent of the literature until quite late in the preparation of this paper. As we expect the method will be unfamiliar to many readers, and as many of the papers concerned contain much very interesting, but for the present purposes distracting, additional material, we will briefly sketch the method. Further examples of its use, and justification, can be found in [18, 3, 8, 9, 6, 16, 30, 27, 31], as well as the already cited early paper of Riordan and Shannon!

In the applications of interest, the number of trees of a certain sort with n leaves will be determined by the (real, nonnegative) coefficient of x^n in an exponential or ordinary generating series

$$T(x) \; = \; T_1 x + T_2 x^2 + \cdots + T_n x^n + \cdots .$$

In each case, $T(x)$ will be the unique formal power series solution of a *functional equation*

$$T(x) = g(x, T(x)). \tag{5}$$

$g(x, y)$ will be given by a power series in x, y, which converges for all complex numbers y, whenever x is a complex number satisfying $|x| < x_1$. Here x_1 is a real number which is sufficiently large so that this disc includes all x values of interest. So if $T(x)$ converges for some such x, then its value will satisfy (5).

For a partial converse, suppose x_0, y_0 are complex numbers (with $|x_0| < x_1$) satisfying

$$y_0 = g(x_0, y_0). \tag{6}$$

Then by the implicit function theorem, the equation

$$y = g(x, y) \tag{7}$$

will have a unique solution $y = T^*(x)$, defined and analytic in some complex neighbourhood of x_0, having $y_0 = T^*(x_0)$, unless $\frac{\partial}{\partial y}(\, g(x, y) - y \,) = 0$ at $x = x_0$, $y = y_0$, i.e., *unless*

$$\frac{\partial g}{\partial y}(x_0, y_0) = 1. \tag{8}$$

In particular, if $\frac{\partial g}{\partial y}(0, 0) \neq 1$, then (7) has a power series solution $y = T^*(x) = T_1^* x + T_2^* x^2 + \cdots$ in some neighbourhood of 0. But since this must satisfy (5) as a formal power series (and the formal power series solution is unique), we see that $T(x) = T^*(x)$ has a positive (possibly infinite) radius of convergence.

Suppose, temporarily, that $T(x)$ has *finite* radius of convergence $x_0 < x_1$, and that $T(x_0)$ converges to $y_0 = \lim_{x\to x_0-} T(x)$. Then x_0, y_0 must satisfy (6) and (8). In our applications these equations will have a *unique* simultaneous real solution, and consequently will determine the only possible values for $x_0, T(x_0)$. From now on we will assume this to be the case, and also that x_0, y_0 is the only complex solution x, y with $|x| \le x_0$.

Suppose additionally, that as well as (6) and (8), x_0, y_0 satisfy

$$\frac{\partial g}{\partial x}(x_0, y_0) \neq 0\,, \quad \frac{\partial^2 g}{\partial y^2}(x_0, y_0) \neq 0\,. \tag{9}$$

Then near x_0, y_0, equation (7) behaves like

$$0 \;=\; \frac{\partial g}{\partial x}(x_0, y_0)\,(x - x_0) \;+\; \frac{\partial^2 g}{\partial y^2}(x_0, y_0)\,\frac{(y-y_0)^2}{2} \;+\; \cdots. \tag{10}$$

From this (or more precisely, from a stronger rigorous version which is commonly called the *Weirstrass preparation theorem*, see for example Markushevich [15]), we see that $T(x)$ does indeed have a singularity (a branch point) at $x = x_0$.

By our assumption that (6) and (8) have only one solution, this is the only singularity $T(x)$ has on its circle of convergence, since for every other x on the circle, $\frac{\partial g}{\partial y}(x, T(x)) \neq 1$, whence $T(x)$ has an analytic continuation in a neighbourhood of x. Using this and the Weirstrass preparation theorem, one can apply *Darboux's lemma* (see for example, [30]) to establish rigorously that asymptotically the coefficients of $T(x)$ behave exactly as we would expect from (10). That is, if we solve

$$(y_0 - y)^2 = 2\,\beta\,(x_0 - x) \quad \text{where} \quad \beta = \frac{\partial g}{\partial x}(x_0, y_0) \Big/ \frac{\partial^2 g}{\partial y^2}(x_0, y_0)\,,$$

as

$$y \;=\; y_0 - \sqrt{2\beta x_0}\,\left(1 - \frac{x}{x_0}\right)^{\frac{1}{2}} \;=\; \sum_{n=0}^{\infty} \alpha_n x^n\,,$$

by expanding $(1 - x/x_0)^{1/2}$ using (say) the binomial theorem, then as $n \to \infty$,

$$T_n \;\sim\; \alpha_n = \sqrt{2\beta x_0}\,\frac{(2n)!\,x_0^{-n}}{2^{2n}(n!)^2(2n-1)} \;\sim\; \sqrt{\frac{\beta x_0}{2\pi}}\;n^{-3/2} x_0^{-n}. \tag{11}$$

4 Read-once functions.

Reduced formulas φ, which come in two varieties $\wedge$-*formulas* and $\vee$-*formulas*, are defined inductively as follows:

(i) Every literal is both an $\wedge$-formula and an $\vee$-formula.

(ii) If Φ is a finite multiset of at least two $\vee$-formulas, then $\wedge\Phi$ is an $\wedge$-formula.

(iii) If Φ is a finite multiset of at least two $\wedge$-formulas, then $\vee\Phi$ is an $\vee$-formula.

Identity of reduced formulas φ, φ' is also defined inductively. $\varphi = \varphi'$ if they are identical literals, or if both are $\wedge$-formulas ($\vee$-formulas), say $\wedge\Phi$, $\wedge\Phi'$ ($\vee\Phi$, $\vee\Phi'$, respectively), and $\Phi = \Phi'$ as multisets.

Rigid formulas, which will be defined formally in Section 5 as formulas having no nontrivial automorphism, are exactly those reduced formulas which can be constructed according to this definition, if in (ii) and (iii), Φ is restricted to being a *set*, rather than a multiset. The read-once formulas defined below will automatically be rigid. As $\wedge, \vee$ are the only connectives we allow, it turns out that for dealing with the formula complexity of functions, considering only rigid formulas suffices, a point which does not seem to have been fully appreciated in the past. However the "standard" enumeration techniques used in Sections 5 and 6, are based on the case where nonrigid formulas are included in the count, and as we shall see, for many purposes their inclusion makes only a small difference.

In common with other sorts of formula, we may visualize a reduced formula as a rooted tree, with a literal at each leaf, and either $\wedge$ or $\vee$ at each non-leaf node. The indegree of each $\wedge, \vee$ is at least 2, and the order of the nodes feeding it is considered irrelevant. All the nodes feeding an $\wedge$ (respectively $\vee$) are literals or $\vee$ nodes ($\wedge$ nodes), so working backwards from the root there are alternate levels of $\wedge$'s and $\vee$'s.

Given any propositional formula φ, one can readily find a reduced formula φ' computing the same function, and having the property that each literal occurs the same number of times in both φ and φ'. Indeed each reduced formula can be thought of as representing a class comprised of all formulas equivalent to each other by virtue of the *commutativity* and *associativity* of $\wedge$ and $\vee$.

Historical note.

A series-parallel network is usually thought of as a particular sort of connected graph joining two distinguished vertices, the *terminals*. For a switching network, each edge is assigned a literal X_i or $\overline{X}_i$, which historically was a relay contact, either normally open or normally closed, controlled by the variable X_i . The network computes a Boolean function $f(X_1, \ldots, X_n)$, with the value of $f(x_1, \ldots, x_n)$ for $x_1, \ldots, x_n \in \{0, 1\}$, determined by whether the terminals are connected by closed contacts. Formally, *two terminal series-parallel networks* come in two varieties, *essentially series* and *essentially parallel*, and following Riordan and Shannon [23] are defined inductively by:

(i) An edge (with a literal) is both an essentially series and an essentially parallel network.

(ii) Given a finite multiset of at least two essentially parallel networks, their "series connection" (without regard to order) is an essentially series network.

(iii) Given a finite multiset of at least two essentially series networks, their "parallel connection" (without regard to order) is an essentially parallel network.

Comparing this with the definition of reduced formula, the reader will immediately see that series-parallel networks with L switches are in one to one correspondence with reduced formulas of size L, and that corresponding networks and formulas compute the same Boolean function. According to Dunne [4], page 271, "the theory of relay-contact circuits is no longer technologically relevant". Although it is probably true that *mechanical* switches are no longer relevant, we are not so sure that the same should be said of series-parallel switching networks in general. Nevertheless, having noted their *equivalence* with reduced formulas, we will continue to be "modern" and use the formula nomenclature.

A Boolean function $f(X_1, \ldots, X_n)$ *depends on* X_n if there is some choice of $x_1, \ldots, x_{n-1} \in \{0, 1\}$ such that $f(x_1, \ldots, x_{n-1}, 0) \neq f(x_1, \ldots, x_{n-1}, 1)$. Dependence on the variable X_i is defined similarly.

$f(X_1, \ldots, X_n)$ is a *read-once function* if it is computed by some formula $\varphi(X_1, \ldots, X_n)$ in which no variable occurs more than once. Obviously, $\varphi(X_1, \ldots, X_n)$ (which is called a *read-once formula*), may be taken to be a reduced formula.

Read-once functions have an extensive literature. For some of the more recent contributions, and many more references, see Angluin, Hellerstein, and Karpinski [1], Eiter [5], and Mundici [17]. The following lemma is well known, being explicitly stated in [1],[5]. (Cf., Hunt and Stearns [10].)

LEMMA 4.1 *Suppose $f(X_1, \ldots, X_n)$ is a read-once Boolean function. Then the reduced formula computing f in which each variable appears at most once, is unique.*

PROOF: Consider a reduced read-once formula $\varphi(X_1, \ldots, X_n)$ computing f. Clearly, if X_n appears unnegated in φ, then $f(x, 0) \leq f(x, 1)$ for all assignments $x \in \{0,1\}^{n-1}$, with strict inequality for some x. Dually, if X_n appears negated, then $f(x, 0) \geq f(x, 1)$ for all assignments $x \in \{0,1\}^{n-1}$, with strict inequatity for some x. Thus the function f determines whether X_n (or similarly any X_i) actually occurs in φ (which it will if and only if f depends on X_n), and whether X_n appears as the literal X_n, or in $\overline{X}_n$.

Therefore we may suppose that f depends on all of $X_1, \ldots, X_n$, that $n > 1$, and that $X_1, \ldots, X_n$ all appear unnegated in φ (i.e., f is monotone). Let us say that X_1 and X_2 are $\wedge$-connected, if there is a setting x' of the remaining variables such that $f(X_1, X_2, x') \equiv X_1 \wedge X_2$. Define X_i and X_j $(i \neq j)$ being $\wedge$-connected, and $\vee$-connected variables analogously.

If f is expressible by a read-once formula, then every two variables are either $\wedge$-connected or $\vee$-connected, and never both.

Consider the transitive (and reflexive) closures of the relations "$\wedge$-connected" and "$\vee$-connected". If φ is an $\vee$-formula, then since the $\vee$ at the root has fan-in at least two, the relation "transitively-$\vee$-connected" holds for every pair of variables. Similarly, the relation "transitively-$\wedge$-connected" is an equivalence relation whose blocks determine the sets of variables in the direct subformulas of the root. (Variables inputting directly to the root $\vee$ form singleton blocks of transitively-$\wedge$-connected.)

Since dual statements, with $\wedge$ and $\vee$ swapped, hold if φ is a $\wedge$-formula, an inductive application of these observations shows that f determines the reduced formula identically.

■

THEOREM 4.2 *The number r_n of read-once Boolean functions in $X_1, \ldots, X_n$ which depend on all these variables, satisfies*

$$r_n \sim \sqrt{\frac{\ln 4 - 1}{\pi}} \frac{c^n}{n^{3/2}} n! \sim \sqrt{2(\ln 4 - 1)} \left(\frac{c}{e}\right)^n n^{n-1},$$

where $c = 2/(\ln 4 - 1)$.

Proof: By Lemma 4.1, r_n is simply the number of reduced formulas in which each variable $X_1, \ldots, X_n$ occurs exactly once (negated or unnegated). Obviously

$$r_n = 2^n r_n^+ ,$$

where r_n^+ is the number of reduced formulas in which only *unnegated* variables $X_1, \ldots, X_n$ are used, and each occurs exactly once. (r_n^+ is also the number of monotone read-once functions depending on exactly $X_1, \ldots, X_n$.) But in a reduced formula, the assignment of connectives $\wedge, \vee$ to the nonleaf nodes is completely determined by which of these symbols is assigned to the root. So for $n \geq 2$,

$$r_n^+ = 2 s_n ,$$

where s_n is the number of nonisomorphic rooted trees having n leaves with distinct labels $X_1, \ldots, X_n$ (or, if you prefer, $1, \ldots, n$), in which each node has in-degree either 0 or at least 2. Let

$$S(x) \;=\; \sum_{n=1}^{\infty} \frac{s_n}{n!}\, x^n$$

be the exponential generating series corresponding to such trees. It is well known (see, e.g., [27],[30]), and not hard to prove, that the exponential generating series for a forest of m trees of this sort, with the forest having n (say) leaves with distinct labels $X_1, \ldots, X_n$, is $(S(x))^m/m!$. Each tree consists of either a single leaf, or a root together with a forest of $m \geq 2$ trees with their roots joined by directed edges to the root. So

$$S(x) \;=\; x + \sum_{m=2}^{\infty} \frac{(S(x))^m}{m!} \;=\; x + e^{S(x)} - S(x) - 1.$$

Notice that the coefficient $s_n/n!$ of x^n on the left hand side, is determined by the coefficients $s_j/j!$ with $j < n$ on the right hand side. Therefore $y = S(x)$ is the unique formal power series solution of the functional equation:

$$y \;=\; \frac{1}{2}\,(x + e^y - 1)\;. \tag{12}$$

Let us apply the method described in Section 3 to determine the properties of $S(x)$. The power series $g(x,y) = \frac{1}{2}\,(x + e^y - 1)$ converges for all complex numbers x, y. Since $\frac{\partial g}{\partial y}(0,0) \neq 1$, the radius of convergence of $S(x)$ is positive. On the other hand, it is easy to see that $s_n \geq n!/2$, so the radius of convergence is at most that of $\sum_n x^n$. The radius of convergence x_0 is therefore certainly finite. Either $S(x_0)$ converges, or as $x \to x_0-$, $S(x) \to \infty$. But whenever $S(x)$ converges, $S(x) - \frac{1}{2}\, e^{S(x)} = \frac{1}{2}\,(x-1)$, so as $x \to x_0-$,

$$S(x) - \frac{1}{2}\; e^{S(x)} \;\to\; \frac{1}{2}\,(x_0 - 1)\,,$$

which is incompatible with $S(x) \to \infty$. Therefore $S(x_0) = y_0$, say, and x_0, y_0 satisfy (6) and (8), that is,

$$\begin{aligned} y_0 &= \frac{1}{2}\,(x_0 + e^{y_0} - 1) \\ \frac{1}{2}\, e^{y_0} &= 1\,. \end{aligned}$$

From these we see immediately that $y_0 = \ln 2$, $x_0 = 2\ln 2 - 1 = \ln 4 - 1$.

Also (9) is satisfied, so by (11),

$$\frac{s_n}{n!} \;\sim\; \sqrt{\frac{x_0}{4\pi}}\;\frac{x_0^{-n}}{n^{3/2}}\,, \qquad \text{where} \;\; x_0 = \ln 4 - 1. \tag{13}$$

Consequently, the numbers r_n^+, r_n are given asymptotically by:

$$\begin{aligned} r_n^+ &\sim \sqrt{\frac{\ln 4 - 1}{\pi}}\;\frac{(c/2)^n}{n^{3/2}}\; n!\,, \\ r_n &\sim \sqrt{\frac{\ln 4 - 1}{\pi}}\;\frac{c^n}{n^{3/2}}\; n!\,, \qquad \text{where} \;\; c = \frac{2}{\ln 4 - 1}\,. \end{aligned}$$

■

Let R_n (respectively R_n^+) be the number of (monotone) read-once functions in $X_1, \ldots, X_n$, including those which depend only on some of these variables. Then

$$\begin{aligned} R_n &= 2n + \sum_{l=2}^{n} \binom{n}{l} r_l \;\sim\; \sum_{j=0}^{\omega(n)} \frac{n^j}{j!}\; r_{n-j} \;\sim\; r_n \sum_{j=0}^{\infty} \frac{c^{-j}}{j!}\,, \\ R_n^+ &= n + \sum_{l=2}^{n} \binom{n}{l} r_l^+ \;\sim\; \sum_{j=0}^{\omega(n)} \frac{n^j}{j!}\; r_{n-j}^+ \;\sim\; r_n^+ \sum_{j=0}^{\infty} \frac{x_0^j}{j!}\,, \end{aligned}$$

for $\omega(n) \to \infty$ sufficiently slowly, and $x_0 = \ln 4 - 1$, yielding:

Corollary 4.3

$$R_n \sim \sqrt{\frac{4}{e}}\, r_n \sim 2\sqrt{\frac{2(\ln 4 - 1)}{e}}\left(\frac{c}{e}\right)^n n^{n-1},$$

$$R_n^+ \sim \frac{4}{e}\, r_n^+ \sim \frac{4\sqrt{2(\ln 4 - 1)}}{e}\,(e(\ln 4 - 1))^{-n}\, n^{n-1}\,.$$

Historical note.

The numbers s_n and the functional equation (12) appear to date from the fourth problem of Schröder [26], who considered the number of ways of bracketing $1, 2, \ldots, n$, if the order of objects within parentheses does not matter, and each pair of matching brackets (including a pair around the whole collection) encloses at least two objects. For example,

$$((1\,(2\,4\,5))\,6\,(3\,7))$$

corresponds to the two reduced monotone read-once formulas

$$(X_1 \vee (X_2 \wedge X_4 \wedge X_5)) \wedge X_6 \wedge (X_3 \vee X_7)$$

$$(X_1 \wedge (X_2 \vee X_4 \vee X_5)) \vee X_6 \vee (X_3 \wedge X_7)\ .$$

(Hence $r_n^+ = 2s_n$ for $n \geq 2$.) Interestingly, Schröder observed that $S(x)$ has radius of convergence $x_0 = 2\ln 2 - 1$, which is, of course, where our constant c comes from! Comtet [3] seems to have been the first to obtain the asymptotic formula (13), after which it appears in a number of papers including Foulds and Robinson [6], in the context of *phylogenetic* (evolutionary) *trees*, and Moon [16] for series-parallel networks. The latter paper includes an excellent survey.

A straightforward application of the *Lagrange Inversion Theorem* (see e.g., [27]) to (12), yields the following exact formula for the Schröder numbers s_n, from which analogous exact formulas for $r_n^+ = 2s_n$, $r_n = 2^{n+1}s_n$, R_n^+ and R_n can readily be derived.

Theorem 4.4 *For $n > 1$,*

$$s_n = \sum_{\substack{j_1 \geq 0, \ldots, j_{n-1} \geq 0 \\ j_1 + 2j_2 + \cdots + (n-1)j_{n-1} = n-1}} \frac{(n + j_1 + j_2 + \cdots + j_{n-1} - 1)!}{j_1!\,(2!)^{j_1}\, j_2!\,(3!)^{j_2} \cdots j_{n-1}!\,(n!)^{j_{n-1}}}\ .$$

PROOF: By (12), $S(x) = \sum_{n=1}^{\infty} s_n x^n/n!$ satisfies the equation $x = h(S(x))$, where $h(y) = 1 + 2y - e^y = y - \sum_{k=2}^{\infty} y^k/k!$. As $h(0) = 0$, $h'(0) \neq 0$, the Lagrange Inversion Theorem asserts that $s_n/(n-1)!$ is the coefficient of y^{n-1} in

$$\begin{aligned}
\left(\frac{y}{h(y)}\right)^n &= \left(1 - \sum_{k=1}^{\infty} \frac{y^k}{(k+1)!}\right)^{-n} \\
&= (1-\theta)^{-n}, \qquad \text{where} \quad \theta = \sum_{k=1}^{\infty} \frac{y^k}{(k+1)!}, \\
&= 1 + n\theta + \frac{n(n+1)}{2!}\,\theta^2 + \cdots + \frac{(n+m-1)!}{(n-1)!\,m!}\,\theta^m + \cdots .
\end{aligned}$$

The coefficient of y^{n-1} in θ^m is

$$\sum_{\substack{j_1 \geq 0, \ldots, j_{n-1} \geq 0 \\ j_1 + 2j_2 + \cdots + (n-1)j_{n-1} = n-1 \\ j_1 + j_2 + \cdots + j_{n-1} = m}} \frac{m!}{j_1!\,j_2!\,\cdots\,j_{n-1}!} \left(\frac{1}{2!}\right)^{j_1} \left(\frac{1}{3!}\right)^{j_2} \cdots \left(\frac{1}{n!}\right)^{j_{n-1}},$$

for $n > 1$, and the result follows.

■

Thus $s_2 = 1$, $s_3 = 4$, $s_4 = 26$, $s_5 = 236, s_6 = 2752, \ldots$. More extensive lists can be found in Comtet [3] (Theorem 1 of which, gives the above formula less explicitly, as a sum of *associated Stirling numbers of the second kind*), and, in the guise of r_n^+, as the diagonal entries of Table 6, page 142, of Riordan [22].

5 A general upper bound for functions.

An upper bound on the number $B(n, L) = b(n, L)^L$ of Boolean functions expressible by formulas of size L, will now be established. An alternative proof (historically our first) of the weaker bound $b(n, L) \leq (c + o(1))n$, can be found in Section 6. With a matching lower bound, either proof will supply the requirements for showing $b(n, L) \sim cn$.

Given a reduced formula φ of size L, a *labelled formula* ϕ is obtained from φ by labelling each of the L occurrences of literals in φ by a distinct numeral from $\{1, \ldots, L\}$. Thought of as a tree, each leaf in ϕ is labelled with a numeral, as well as having one of the literals $X_1, \overline{X}_1, \ldots, X_n, \overline{X}_n$ attached. It is easily seen that the number of labelled reduced formulas of size L in which only the variables $X_1, \ldots, X_n$ occur, is exactly

$$n^L\, r_L\ ,$$

where r_L is the number of read-once formulas in L variables, found in Section 4.

A permutation α of the numeral labels $\{1, \ldots, L\}$ is an *automorphism* of a labelled formula ϕ, if it preserves ϕ. A labelled formula ϕ is *rigid* if its only automorphism is the identity permutation. Similarly, a reduced formula φ is *rigid* if some (or equivalently, every) labelled formula obtained from it is rigid.

LEMMA 5.1 *Let $n \geq 2$, and let $f(X_1, \ldots, X_n)$ be any Boolean function. Then f is computed by some rigid formula of size L in $X_1, \ldots, X_n$, for every $L \geq \max\{\mathbf{L}(f), 3\}$.*

PROOF: Suppose first that $\mathbf{L}(f) \geq 2$. Let φ be a reduced formula computing f of the minimal size $\mathbf{L}(f)$. Then φ is rigid.

To see this, let ϕ be a labelled formula obtained from φ, and suppose α is an automorphism of ϕ which maps m_1 to $m_2 \neq m_1$. Consider the minimal subformula containing these labels. Suppose it is an $\wedge$-formula (the $\vee$-formula-case is dual) of the form $\phi_1 \wedge \phi_2 \wedge \cdots$, where the order of the conjuncts is immaterial. If m_1 is in ϕ_1 and m_2 is in ϕ_2, then ϕ_2 must be the image of ϕ_1 under α. But this entails that ϕ_1 and ϕ_2 with the labels removed are identical reduced formulas, so the formula obtained by omitting one of them still computes f, contrary to the minimality of φ.

Starting with φ, we will include new literals one by one, in such a way that the function computed does not change, and the formula remains rigid. Letting X_i', X_i'' denote literals from $\{X_i, \overline{X}_i\}$, another invariant of the construction is that in every subformula of the form $X_i' \wedge X_j'' \wedge \cdots$ (dually $X_i' \vee X_j'' \vee \cdots$), the variables X_i and X_j are distinct. By minimality, φ satisfies this invariant. If φ' is rigid and satisfies the invariant, find a subformula of the form $X_i' \wedge X_j'' \wedge \cdots$ or $X_i' \vee X_j'' \vee \cdots$ at maximum depth. (This is possible since the size is at least 2 by the assumption $\mathbf{L}(f) \geq 2$.) Replace the subformula by $X_i' \wedge (\overline{X_i'} \vee X_j'') \wedge \cdots$ or $X_i' \vee (\overline{X_i'} \wedge X_j'') \vee \cdots$, respectively. Clearly the reduced formula resulting after cancellation of any double negations, remains rigid, satisfies the invariant, and computes the same function.

If f is the constant function 0 (1 is dual), the lemma is obvious for $L = 3$ (and even for $L = 2$), while for $L \geq 4$ the construction is the same, except that in place of φ we start with $X_1 \wedge X_2 \wedge (\overline{X}_1 \vee \overline{X}_2)$. If f can be defined by a literal, for example $f \equiv X_1'$, the construction works, with φ replaced by $X_1' \vee (X_1' \wedge X_2)$, for $L \geq 3$.

■

THEOREM 5.2 *Let F'_{nL} be the number of (unlabelled) rigid reduced formulas of size L with variables form among $X_1, \ldots, X_n$. Then for $L \geq 3, n \geq 2$,*

$$B(n,L) \;\leq\; F'_{nL} \;\leq\; \frac{n^L r_L}{L!} \;\sim\; \sqrt{\frac{\ln 4 - 1}{\pi}}\;\frac{(cn)^L}{L^{3/2}} \quad \textit{as } L \to \infty.$$

PROOF: By Lemma 5.1, $B(n,L) \leq F'_{nL}$. Clearly every rigid reduced formula φ of size L gives rise to $L!$ distinct labelled formulas ϕ, so $F'_{nL}\,L!$ is at most the number of labelled formulas of size L, that is, $F'_{nL}\,L! \leq n^L\,r_L$. The rest follows by Theorem 4.2.

■

COROLLARY 5.3 *There is a constant l_0 (independent of n) such that for all $L \geq l_0$,*

$$b(n,L) \;<\; cn\,.$$

6 The number of reduced formulas.

Let F_{nL} be the number of reduced formulas with exactly L occurrences of variables from $X_1, \ldots, X_n$. In this section (which apart from Corollary 6.3 — already a consequence of Corollary 5.3 — is not required for the later sections) it is shown that if $n \to \infty,\ L \to \infty$, then F_{nL} has "growth rate"

$$F_{nL}^{1/L} \;\sim\; cn\,, \qquad \text{where } c = 2/(\ln 4 - 1).$$

Every propositional formula is equivalent to a reduced formula with the same number of occurrences of variables, so obviously $B(n,L) \leq F_{nL}$. Indeed for $L \geq 3$, $n \geq 2$,

$$B(n,L) \;\leq\; F'_{nL} \;\leq\; \frac{n^L\,r_L}{L!} \;\leq\; F_{nL}\,.$$

In later sections, it will "come out in the wash", that provided L is not too large, approximating $B(n,L) = b(n,L)^L$ by F_{nL} already gives an asymptotically correct answer for $b(n,L)$, namely,

$$b(n,L) \;\sim\; (F_{nL})^{1/L} \;\sim\; cn.$$

Enroute to this general result for F_{nL}, an asymptotic formula for F_{kL} when k is fixed and $L \to \infty$, will be found using the method of Section 3. To deal with the technicality of showing that $\sum_L F_{kL}\, x^L$ has positive radius of convergence, we will compare F_{kL} with the larger (or equal) number F^*_{kL} of formulas (like (4)) which are written on a line, use only *binary* connectives $\wedge, \vee$, and have L occurrences of literals from among $X_1, \overline{X}_1, X_2, \overline{X}_2, \ldots, X_k, \overline{X}_k$. All that is needed is that $F^*_{kL} \le \alpha_k^L$ for some constant α_k depending only on k, which is easily seen by regarding such formulas as strings (with length linearly bounded by L) over a finite alphabet. But in fact, it is well known (see, e.g., [30],[19] or [11]), that the numbers F^*_{kL} are closely related to the *Catalan numbers* $b_n = \binom{2n}{n}/(n+1)$. They are given exactly by:

$$F^*_{kL} \;=\; (2k)^L\; 2^{L-1}\, b_{L-1} \;\sim\; \frac{(16k)^L}{8\,\sqrt{\pi}\; L^{3/2}} \quad \text{as} \;\; L \to \infty. \tag{14}$$

LEMMA 6.1 *For fixed* k*,*

$$F_{kL} \;\sim\; \kappa_k\, \frac{x_k^{-L}}{L^{3/2}} \qquad \text{as } L \to \infty,$$

where κ_k, x_k *are constants depending only on* k*. Moreover,*

$$x_k^{-1} \;\sim\; ck \qquad \text{as} \;\; k \to \infty.$$

PROOF: In a reduced formula, the assignment of connectives $\wedge, \vee$ to the nonleaf nodes is completely determined by which of these symbols is assigned to the root, so

$$F_{kL} = 2\, T_{kL}\,,$$

where T_{kL} is the number of (nonisomorphic) rooted trees in which each node has in-degree either 0 or at least 2, and there are exactly L leaves, each having one of $2k$ labels $X_1, \overline{X}_1, X_2, \overline{X}_2, \ldots, X_k, \overline{X}_k$. Consider the corresponding ordinary generating series

$$T_k(x) \;=\; \sum_{L=1}^{\infty} T_{kL}\, x^L\,.$$

By a standard argument (e.g., using *Polya's Enumeration Theorem* applied to the symmetric group S_m — see e.g., [8, 18, 20, 21, 22, 27]; or directly — see e.g., [16, 23]), the ordinary generating series for a forest of m such trees is

$$\sum_{\substack{j_1,\ldots,j_m \\ j_1+2j_2+\cdots+mj_m=m}} \frac{(T_k(x))^{j_1}}{j_1!}\,\frac{(T_k(x^2))^{j_2}}{2^{j_2}\,j_2!}\cdots\frac{(T_k(x^m))^{j_m}}{m^{j_m}\,j_m!}\,.$$

As each tree consists of either a single leaf, having one of $2k$ literals as a label, or a root together with a forest of $m \geq 2$ trees with their roots joined by directed edges to the root, it follows that

$$\begin{aligned} T_k(x) &= 2kx + \sum_{m=2}^{\infty} \sum_{\substack{j_1,\ldots,j_m \\ j_1+2j_2+\cdots+mj_m=m}} \frac{(T_k(x))^{j_1}}{j_1!}\,\frac{(T_k(x^2))^{j_2}}{2^{j_2}\,j_2!}\cdots\frac{(T_k(x^m))^{j_m}}{m^{j_m}\,j_m!} \\ &= 2kx + \ e^{\sum_{m=1}^{\infty} T_k(x^m)/m} - 1 - T_k(x)\,. \end{aligned}$$

So $y = T_k(x)$ satisfies the functional equation

$$y \ = \ 2kx + \ e^y \ e^{\sum_{m=2}^{\infty} T_k(x^m)/m} - 1 - y\,. \tag{15}$$

Treating $T_k(x^m)$, $m \geq 2$, as given, it is easy to see that this has a unique formal power series solution $y = T^*(x) = \sum_{L=1}^{\infty} T_L^*\, x^L$, as the coefficient T_L^* of x^L on the left hand side of (15) is determined by T_j^*, $j < L$ on the right. Of course it must be that $T^*(x) = T_k(x)$, so $y = T_k(x)$ is the unique formal power series solution of

$$y \ = \ kx + \ \frac{1}{2}\left(e^y \ e^{\sum_{m=2}^{\infty} T_k(x^m)/m} - 1\right). \tag{16}$$

In order to apply the method of Section 3, we must check that $T_k(x)$ has positive radius of convergence. But $T_{kL} \leq F_{kL} \leq F_{kL}^*$, so $T_k(x)$ has radius of convergence at least as great as that of $\sum_L F_{kL}^*\, x^L$, which by (14) is $(16k)^{-1}$.

Suppose x_k is a real number, $0 < x_k < 1$, and that $T_k(x)$ converges for $0 \leq x < x_k$. Then certainly $T_k(x_k^m)$ converges for each $m \geq 2$. Also,

$$\sum_{m=3}^{\infty} \frac{T_k(x^m)}{m} \ \leq \ \sum_{m=1}^{\infty} \frac{x^m\, T_k(x^2)}{m+2},$$

so $\sum_{m=2}^{\infty} T_k(x^m)/m$ converges to $\sum_{m=2}^{\infty} T_k(x_k^m)/m$ as $x \to x_k-$. Therefore

$$T_k(x) - \frac{1}{2} e^{T_k(x)} e^{\sum_{m=2}^{\infty} T_k(x^m)/m} = kx - \frac{1}{2} \to kx_k - \frac{1}{2}$$

as $x \to x_k-$, so

$$T_k(x) - \frac{1}{2} e^{T_k(x)} e^{\sum_{m=2}^{\infty} T_k(x_k^m)/m} \to kx_k - \frac{1}{2} .$$

But the latter is incompatible with $T_k(x) \to \infty$, so $T_k(x_k) = y_k$, say. Clearly,

$$y_k = kx_k + \frac{1}{2}\left(e^{y_k} e^{\sum_{m=2}^{\infty} T_k(x_k^m)/m} - 1 \right). \tag{17}$$

It follows that if $T_k(x)$ has radius of convergence $x_k < 1$, then x_k, $y_k = T_k(x_k)$, must satisfy both (17) and

$$\frac{1}{2} e^{y_k + \sum_{m=2}^{\infty} T_k(x_k^m)/m} = 1 . \tag{18}$$

These equations are just (6) and (8) (with x_k, y_k in place of x_0, y_0), if we take

$$g(x,y) = kx + \frac{1}{2}\left(e^{y + \sum_{m=2}^{\infty} T_k(x^m)/m} - 1 \right).$$

Let us consider equations (17) and (18) *without* assuming that the radius of convergence is less than 1. They easily reduce to

$$\begin{aligned} y_k &= \ln 2 - \sum_{m=2}^{\infty} \frac{T_k(x_k^m)}{m} \\ x_k &= \frac{1}{k}\left(c^{-1} - \sum_{m=2}^{\infty} \frac{T_k(x_k^m)}{m} \right) . \end{aligned}$$

As $c^{-1} < 1$, and $x + \frac{1}{k}\sum_{m=2}^{\infty} T_k(x^m)/m$ is monotone increasing for real $x > 0$, it is clear that these equations have exactly one positive solution x_k, y_k. Moreover, $x_k < (kc)^{-1} < 1/k$, and the radius of convergence of $T_k(x)$ is at least x_k. But (9) holds for x_k, y_k, so $T_k(x)$ does indeed have a singularity there, and therefore the radius of convergence is x_k. Also, if x is a singularity on the circle $|x| = x_k$, then by (18), and its analogue for x,

$$T_k(x) + \sum_{m=2}^{\infty} \frac{T_k(x^m)}{m} \;=\; \ln 2 \;=\; T_k(x_k) + \sum_{m=2}^{\infty} \frac{T_k(x_k^m)}{m}\,.$$

As all the coefficients of $T_k(x)$ are strictly positive, this is only possible if $x = x_k$. Therefore (11) can be applied to yield:

$$F_{kL} \;=\; 2\,T_{kL} \;\sim\; \kappa_k\,\frac{x_k^{-L}}{L^{3/2}}\,, \qquad \text{for some constant } \kappa_k.$$

Finally, since $x_k < 1/k$,

$$\sum_{m=2}^{\infty} \frac{T_k(x_k^m)}{m} \;\le\; \sum_{m=1}^{\infty} \frac{x_k\,T_k(x_k^m)}{m+1} \;<\; \frac{1}{k}\,\sum_{m=1}^{\infty} \frac{T_k(x_k^m)}{m} \;=\; \frac{\ln 2}{k}\,.$$

Therefore, as $k \to \infty$,

$$x_k = \frac{1}{k}\left(c^{-1} - \sum_{m=2}^{\infty} \frac{T_k(x_k^m)}{m}\right) \;\sim\; (ck)^{-1}\,.$$

■

Historical note.

$T_k(x)$ is the ordinary generating series for rooted trees (counted by leaves) in which each leaf has one of $2k$ labels, and no vertex has indegree 1 (when the edges are directed towards the root). Clearly the above argument works for $k = \frac{1}{2}, 1, \frac{3}{2}, 2, \frac{5}{2}, \ldots$. (Another interpretation of F_{kL} for $k = n/2$, is that F_{kL} is the number of *monotone* reduced formulas of size L in n variables.) In the special case $k = \frac{1}{2}$ of indistinguishable leaves, the generating series was considered by Cayley [2] (in the context of trees), and MacMahon [14] (for series-parallel networks). The asymptotic formula (for $k = \frac{1}{2}$) is proved explicitly in Moon [16], its form having been more or less conjectured by Riordan and Shannon [23], who seem to have initially missed the relevance of Polya [20]. For general k, the functional equation (16) already appears implicitly in Riordan [22], problem 31, page 159.

It follows immediately from the lemma, that there is *some* function $l(k)$ such that if $L \ge l(k)$, then

$$F_{kL}^{1/L} \;\le\; x_k^{-1} \;=\; ck + o(k)\,. \tag{19}$$

In fact, to ensure $F_{kL}^{1/L} \le ck + o(k)$, *any* function $l(k) \to \infty$ will do.

Theorem 6.2 *Let $l(n)$ be any function with the property that $l(n) \to \infty$ with n. Then as $n \to \infty$,*

$$F_{nL}^{1/L} \sim cn \ ,$$

uniformly for all $L \geq l(n)$.

Proof: For the upper bound on $F_{nL}^{1/L}$, we must show that for any $\epsilon > 0$, if n is sufficiently large then for all $L \geq l(n)$, $F_{nL}^{1/L} \leq (c+\epsilon)\, n$. By (19), there is some k such that $x_k^{-1} \leq (c+\epsilon/2)\, k$. Fix on such a k, and partition the variables $X_1, \ldots, X_n$ into k disjoint subsequences $\mathbf{X}_1, \ldots, \mathbf{X}_k$, each having either $[n/k]$ or $[n/k]+1$ elements. Consider any reduced formula φ in k variables $X_1, \ldots, X_k$. For $i = 1, \ldots, k$, at each occurrence of X_i in φ, choose some variable from the sequence $\mathbf{X}_i$, and replace X_i by it. Clearly the result is a reduced formula in the variables $X_1, \ldots, X_n$, and every reduced formula in $X_1, \ldots, X_n$ can be obtained from some φ in this way. (The original variables $X_1, \ldots, X_k$ are simply being used as place markers for the k sorts of variable which make up $X_1, \ldots, X_n$.) As this process does not change the number L of occurrences of variables in φ, and for each occurrence one of at most $[n/k]+1$ possibilities is chosen, it follows immediately that

$$F_{nL} \leq \left(\frac{n}{k} + 1\right)^L F_{kL} \ .$$

So from (19), and our choice of k so that $x_k^{-1} \leq (c+\epsilon/2)\, k$, we see that

$$F_{nL}^{1/L} \leq \left(\frac{n}{k} + 1\right) F_{kL}^{1/L} \leq \left(\frac{n}{k} + 1\right) x_k^{-1} \leq (c+\epsilon)\, n,$$

provided n (and therefore L, since $L \geq l(n)$) is sufficiently large.

For the lower bound, notice that every reduced formula φ of size L corresponds to at most $L!$ labelled formulas ϕ. As observed in Section 5, the number of labelled formulas of size L is $n^L\, r_L$, so $L!\, F_{nL} \geq n^L\, r_L$. Therefore, using Theorem 4.2,

$$F_{nL}^{1/L} \geq n \left(\frac{r_L}{L!}\right)^{1/L} \sim nc.$$

■

As $b(n,L) \leq F_{nL}^{1/L}$, the upper bound gives another proof of:

Corollary 6.3 *If $l(n) \to \infty$ with n, then uniformly for $L \geq l(n)$,*

$$b(n,L) \leq (c + o(1))\, n \, .$$

7 Small formulas.

We will now complete a proof that for any function $L = l(n)$, $b(n, L) \sim cn$, provided only that $l(n) \to \infty$, and $l(n) = 2^{o(n)}$.

$\mathbf{X}$ will represent a sequence of variables $X_1, \ldots, X_m$. $|\mathbf{X}|$ will denote the length of this sequence. (In this example, $|\mathbf{X}| = m$.) x will denote an assignment of values $x \in \{0,1\}^{|\mathbf{X}|}$ to these variables $\mathbf{X}$. Similarly if $\mathbf{Y}, \mathbf{Z}$ are sequences of variables, y, z will represent assignments of length $|\mathbf{Y}|, |\mathbf{Z}|$, respectively. For any assignment x of length $|\mathbf{X}|$, we will let $\mathbf{X}^x$ denote the unique conjunction of $|\mathbf{X}|$ different literals, with variables from $\mathbf{X}$, which takes value 1 at x. If $f(\mathbf{X}, \mathbf{Y})$ is a function of variables $\mathbf{X}, \mathbf{Y}$, then $f(x, \mathbf{Y})$ will denote the Boolean function of the variables $\mathbf{Y}$ obtained by assigning the values x to the variables $\mathbf{X}$. The notation $f(\mathbf{X}) \equiv f'(\mathbf{X})$ means that the Boolean functions f, f' are identically equal, i.e., that any formulas defining f, f' are equivalent. In particular, $f(x, \mathbf{Y}) \equiv f'(\mathbf{Y})$ means that these are identically equal *as functions of* $\mathbf{Y}$. We adopt the convention $\mathbf{L}(f) = 0$ if f is one of the constant functions $f \equiv 1$, $f \equiv 0$.

We shall use the following (known) technical lemma.

Lemma 7.1 *(The "Combining Lemma".) Suppose $f(\mathbf{X}, \mathbf{Y})$ is a Boolean function with variables $\mathbf{X}, \mathbf{Y}$. Then*

$$\mathbf{L}(f(\mathbf{X}, \mathbf{Y})) \leq \sum_y \mathbf{L}(f(\mathbf{X}, y)) + 2^{|\mathbf{Y}|+1} - 2.$$

Proof: By induction on $|\mathbf{Y}|$. At the induction step, use the fact that

$$f(\mathbf{X}, Y_0, \mathbf{Y}) \equiv \Big(Y_0 \wedge f(\mathbf{X}, 1, \mathbf{Y})\Big) \vee \Big(\overline{Y_0} \wedge f(\mathbf{X}, 0, \mathbf{Y})\Big) .$$

∎

Theorem 7.2 *Suppose $l_1(n) \to \infty$, and $l_2(n) = o(n)$. Then*

$$b(n, L) \geq (c - o(1))\, n ,$$

uniformly for all L satisfying $l_1(n) \leq L \leq 2^{l_2(n)}$.

Proof: Recall that by Theorem 4.2, the number r_l of read-once functions depending on exactly l variables satisfies

$$r_l \sim \kappa \frac{c^l}{l^{3/2}}\, l! \geq (c - o(1))^l\, l! \qquad \text{as } l \to \infty, \tag{20}$$

for some constant κ.

Let $l_3(n)$ be any function satisfying $l_3(n) = o(n)$ and $l_3(n)/\log n \to \infty$. We consider the possibilities $l_1(n) \le L \le l_3(n)$ and $l_3(n) \le L \le 2^{l_2(n)}$ separately.

Suppose $l_1(n) \le L \le l_3(n)$. Then considering only functions defined by read-once formulas with L literals, we see that

$$B(n, L) \ \ge\ \binom{n}{L} r_L \ \ge \frac{(n-L)^L}{L!}\, r_L \ \ge\ \Big((c - o(1))(n - L)\Big)^L = \Big((c - o(1))\, n\Big)^L,$$

so $b(n, L) \ge (c - o(1))\, n$.

Now, suppose $l_3(n) \le L \le 2^{l_2(n)}$. Partition the n variables into disjoint subsequences $\mathbf{X}, \mathbf{Y}$ with $|\mathbf{X}| + |\mathbf{Y}| = n$, where $|\mathbf{Y}| = [\log(L/l_3(n))]$. Clearly, $|\mathbf{Y}| \le l_2(n) = o(n)$ and hence $|\mathbf{X}| \sim n$. Moreover, let l be determined by the inequality

$$(l + 2)\, 2^{|\mathbf{Y}|} \ \le\ L \ <\ (l + 3)\, 2^{|\mathbf{Y}|}.$$

Clearly, $l = \Theta(l_3(n))$, since $l_3(n) - 3 < l < 2\, l_3(n)$.

Now, let us describe the family of functions $f(\mathbf{X}, \mathbf{Y})$ which will turn out to be large enough to give us the required lower bound.

For each assignment $y \in \{0, 1\}^{|\mathbf{Y}|}$, we choose $f(\mathbf{X}, y)$ to be a read-once function depending on exactly l variables from $\mathbf{X}$. Then $\mathbf{L}(f(\mathbf{X}, y)) = l$, so by the combining lemma,

$$\mathbf{L}(f) \ \le\ l\, 2^{|\mathbf{Y}|} + 2^{|\mathbf{Y}|+1} \ =\ (l + 2)\, 2^{|\mathbf{Y}|} \ \le\ L.$$

Also, for each y, since $l \to \infty$, the number of possible choices of $f(\mathbf{X}, y)$ is

$$\binom{|\mathbf{X}|}{l} r_l \ \ge \frac{(|\mathbf{X}| - l)^l}{l!}\, r_l \ \ge\ \Big((c - o(1))\, n\Big)^l,$$

by (20). Bounding $B(n, L)$ by the number of distinct functions $f(\mathbf{X}, \mathbf{Y})$ of the sort just described, shows

$$B(n, L) \ \ge\ \Big((c - o(1))\, n\Big)^{l\, 2^{|\mathbf{Y}|}} \ge\ \Big((c - o(1))\, n\Big)^{L\,(1 - 3/l)}.$$

As $l = \Theta(l_3(n))$, we have $1/l = o(1/\log n)$. Hence

$$\log b(n, L) \ \ge\ \Big(1 - \frac{3}{l}\Big)\Big(\log n + \log c - o(1)\Big) \ \ge\ \log n + \log c - o(1)\,.$$

■

8 Medium sized formulas.

In order to extend the lower bound on $B(n, L)$ to larger values of L, we will partition the set of assignments x. The idea is similiar to "hashing" — with a suitable (e.g., "random") choice of partition, distinct functions, of a simple sort, will remain distinct on almost all the blocks of the partition.

LEMMA 8.1 *Consider any constant $\beta < (\log e)/2$, and let H be a set with $l\,|H| \leq \beta\, 2^{n-2l}/\log n$, where $l > 1$ is an integer. Then there exists a collection $\mathcal{X}_h$, $h \in H$, indexed by H, of disjoint subsets $\mathcal{X}_h \subseteq \{0,1\}^n$ with the following properties:*

(i) For every pair of distinct functions $f_1(\mathbf{X}), f_2(\mathbf{X})$, each depending only on l variables from among $\mathbf{X} = X_1, \ldots, X_n$, and every $h \in H$, there is some $x \in \mathcal{X}_h$ for which $f_1(x) \neq f_2(x)$.

(ii) $|\mathcal{X}_h| \leq n^{2l}$ for all $h \in H$.

PROOF: Consider a random colouring of the n dimensional "cube" $\{0,1\}^n$ in which every element receives exactly one of the "colours" $h \in H$. Every pair $f_1(\mathbf{X}), f_2(\mathbf{X})$ of distinct functions depending on only l variables each, differ everywhere on some $n - 2l$ dimensional "subcube", obtained by setting the essential variables of both f_1 and f_2 in such a way that $f_1(x)$ and $f_2(x)$ become distinct constants. We will show that with probability greater than 0, all except $o(|H|)$ of the colours has the property, that *every* one of these subcubes includes an element of that colour.

The probability that of the 2^{n-2l} points in the subcube, none has colour h, is

$$\left(1 - \frac{1}{|H|}\right)^{2^{n-2l}}.$$

Therefore the probability μ that colour h is missing from at least one of the $\binom{n}{2l} 2^{2l}$ subcubes, is at most

$$\binom{n}{2l} 2^{2l} \left(1 - \frac{1}{|H|}\right)^{2^{n-2l}} \;\leq\; n^{2l}\, e^{-2^{n-2l}/|H|},$$

so

$$\log \mu \;\leq\; 2l \log n - \frac{2^{n-2l}}{|H|} \log e \;\leq\; (2 - (\log e)/\beta)\; l \log n \;\rightarrow\; -\infty.$$

Therefore the expected number of "bad" colours is $\mu|H| = o(|H|)$, as $n \to \infty$, and there will certainly exist a colouring with at most this number. To see that this suffices for property (i), observe that as n is assumed to be sufficiently large, we could instead have started with a set of colours H' having $\beta' 2^{n-2l}/\log n < l\,|H'| \le \beta'' 2^{n-2l}/\log n$ for some β', β'' satisfying $\beta < \beta' < \beta'' < (\log e)/2$. Then $|H'| - o(|H'|)$ of the colours would have the required property, and as $|H|$ is smaller than this, $|H|$ of these monochromatic blocks could be indexed as $\mathcal{X}_h$, $h \in H$.

Then for each $h \in H$, the set $\mathcal{X}_h$ includes at least one point from each of the $\binom{n}{2l} 2^{2l}$ subcubes. But clearly $\mathcal{X}_h$ has a subset $\mathcal{X}'_h$ of cardinality $|\mathcal{X}'_h| \le \binom{n}{2l} 2^{2l} \le n^{2l}$ with the same property. This choice of $\mathcal{X}'_h$, $h \in H$ satisfies both (i) and (ii).

■

Theorem 8.2 *There is a constant $\alpha > 1$ such that for every real $\kappa > 0$, if n is sufficiently large, then*

$$b(n, L) \;\ge\; \alpha^{-1/\kappa}\, cn\,,$$

for all L satisfying $n \le L \le 2^n/n^\kappa$.

Proof: Given the constant κ, choose l to be the integer satisfying

$$2^{2l} \log n \;\le\; \beta n^\kappa \;<\; 2^{2l+1} \log n\,,$$

where β is a positive constant as in Lemma 8.1. We may then suppose that $L \ge n^{4l}$, since $n^{4l} = 2^{o(n)}$ so Theorem 7.2 implies that $b(n, L) \ge (c - o(1))\, n$ for $L \le n^{4l}$.

Partition the n variables into a pair of disjoint sequences $\mathbf{X}, \mathbf{Y}$ with $|\mathbf{X}| + |\mathbf{Y}| = n$, and $|\mathbf{Y}| = [3l \log n]$. Let $\mathcal{X}_h$, $h \in H$, be a collection of disjoint subsets of $\{0,1\}^{|\mathbf{X}|}$ with the properties guaranteed by Lemma 8.1, applied with $|\mathbf{X}|$ in place of n. We will choose $|H|$ later. For each $h \in H$, let $\mathcal{X}_h(\mathbf{X})$ denote the characteristic function of the set $\mathcal{X}_h$.

Consider the family of functions $f(\mathbf{X}, \mathbf{Y})$ of the form

$$\begin{aligned} f(\mathbf{X}, \mathbf{Y}) \;&\equiv\; \bigvee_{h \in H} \Big(f'_h(\mathbf{X}, \mathbf{Y}) \;\wedge\; \mathcal{X}_h(\mathbf{X}) \Big) \\ &\equiv\; \bigvee_{h \in H} \Big(f'_h(\mathbf{X}, \mathbf{Y}) \;\wedge\; \bigvee_{x \in \mathcal{X}_h} \mathbf{X}^x \Big)\,, \qquad (21) \end{aligned}$$

where for each $y \in \{0,1\}^{|\mathbf{Y}|}$, $f'_h(\mathbf{X}, y)$ is a read-once function depending on exactly l of the variables in $\mathbf{X}$. By the Combining Lemma (Lemma 7.1),

$$\mathbf{L}(f'_h(\mathbf{X},\mathbf{Y})) \;\le\; \sum_y \mathbf{L}(f'_h(\mathbf{X},y)) \;+\; 2^{|\mathbf{Y}|+1} \;=\; 2^{|\mathbf{Y}|}(l+2)\,,$$

so it is apparent from (21), together with the inequality $n^{2l+1} < 2^{|\mathbf{Y}|}$, that

$$\begin{aligned}\mathbf{L}(f(\mathbf{X},\mathbf{Y})) \;&\le\; \sum_{h\in H}\Big(\mathbf{L}(f'_h(\mathbf{X},\mathbf{Y})) \;+\; |\mathcal{X}_h|\,|\mathbf{X}|\Big)\\ &\le\; |H|\,\Big(2^{|\mathbf{Y}|}(l+2)+n^{2l}n\Big) \;\le\; |H|\,2^{|\mathbf{Y}|}(l+3).\end{aligned}$$

We make $\mathbf{L}(f) \le L$ by choosing $|H|$ to be the integer determined by

$$|H|\,2^{|\mathbf{Y}|}(l+3) \;\le\; L \;<\; (|H|+1)\,2^{|\mathbf{Y}|}(l+3)\,.$$

Observe that provided $L \le 2^n/n^{\kappa}$,

$$l\,|H| \;\le\; \frac{L}{2^{|\mathbf{Y}|}} \;\le\; \frac{2^{n-|\mathbf{Y}|}}{n^{\kappa}} \;\le\; \beta\,\frac{2^{|\mathbf{X}|-2l}}{\log n}\,,$$

so it really is possible to apply Lemma 8.1 to find a collection with this cardinality $|H|$. Also, $|H|/\log n \to \infty$ (since $L \ge n^{4l}$).

By Lemma 8.1, each of the $\binom{|\mathbf{X}|}{l}\,r_l$ possible choices of a read-once function $f'_h(\mathbf{X}, y)$ leads to a different function $f'_h(\mathbf{X}, y) \wedge \mathcal{X}_h(\mathbf{X})$. So considering only functions f of the form (21), we see that

$$B(n,L) \ge \left(\binom{|\mathbf{X}|}{l}\,r_l\right)^{|H|\,2^{|\mathbf{Y}|}} \ge \left(\frac{(|\mathbf{X}|-l)^l}{l!}\,r_l\right)^{|H|\,2^{|\mathbf{Y}|}} = \left(\frac{r_l}{l!}(n-o(n))^l\right)^{|H|\,2^{|\mathbf{Y}|}}.$$

But by Theorem 4.2, $r_l/l! \ge (c-o(1))^l$, so it follows that

$$\begin{aligned}\log b(n,L) \;&\ge\; \frac{l\,|H|\,2^{|\mathbf{Y}|}}{L}\,(\log n + \log c - o(1))\\ &\ge\; \left(1-\frac{3}{l}\right)\left(1-\frac{1}{|H|}\right)(\log n + \log c - o(1))\\ &\ge\; \log n + \log c - \frac{3\log n}{l} - o(1) \;=\; \log n + \log c - 6/\kappa - o(1)\,,\end{aligned}$$

because $L \le (|H|+1)\,2^{|\mathbf{Y}|}(l+3)$, $|H|/\log n \to \infty$, and $l = (\kappa/2 - o(1))\,\log n$. ∎

Considering $\kappa \to \infty$, this has the immediate corollary:

THEOREM 8.3 *Suppose that for every κ, $l(n) \geq n^\kappa$ when n is sufficiently large. Then*

$$b(n, L) \;\geq\; (c - o(1))\, n\,,$$

uniformly for all L satisfying $n \leq L \leq 2^n/l(n)$.

9 Large formulas.

Determining $b(n, \mathbf{L}_{\max}(n))$ and finding $\mathbf{L}_{\max}(n)$ are equivalent problems. For as $B(n, \mathbf{L}_{\max}(n)) = 2^{2^n}$,

$$\mathbf{L}_{\max}(n) \;=\; \frac{\log B(n, \mathbf{L}_{\max}(n))}{\log b(n, \mathbf{L}_{\max}(n))} \;=\; \frac{2^n}{\log b(n, \mathbf{L}_{\max}(n))}\,.$$

Upper bounds on $\mathbf{L}_{\max}(n)$ can also be used to obtain good lower bounds on $b(n, L)$ for L large.

A set $\mathcal{X}$ of assignments x of length $|\mathbf{X}|$ will be said to be *r-special* if for every $x \in \mathcal{X}$ there is some conjunction $\delta_x(\mathbf{X})$ of r literals (with variables chosen from among $\mathbf{X}$) such that

$$\delta_x(x') = \begin{cases} 1 & \text{if } x = x', \\ 0 & \text{for all other } x' \in \mathcal{X}. \end{cases} \tag{22}$$

The following lemma contains the germ of Lupanov's idea for estimating $\mathbf{L}_{\max}(n)$ from above, and generalizes it.

LEMMA 9.1 *Suppose $f(\mathbf{X}, \mathbf{Y})$ is a Boolean function of $n = |\mathbf{X}| + |\mathbf{Y}|$ variables, with $|\mathbf{Y}| > 4(\log r + \log\log n)$, and that $\mathcal{X}$ is an r-special set of assignments of length $|\mathbf{X}|$. Then there is some $f'(\mathbf{X}, \mathbf{Y})$ such that*

$$f'(x, \mathbf{Y}) \equiv f(x, \mathbf{Y}) \qquad \textit{for all } x \in \mathcal{X},$$

and

$$\mathbf{L}(f') \;\leq\; \frac{r\,|\mathcal{X}|\,2^{|\mathbf{Y}|}}{\log|\mathcal{X}| - \log\log n}\left(1 + \mathrm{O}\left(\frac{1}{r\log n}\right)\right)\,.$$

PROOF: Rename the variables $\mathbf{Y}$ as $\mathbf{Y}, \mathbf{Z}$, where $|\mathbf{Z}|$ will be chosen later. Partition the set of all $2^{|\mathbf{Z}|}$ possible assignments z into at most $2^{|\mathbf{Z}|}/s + 1$ sets S, each having cardinality $|S| \le s$. (s will be chosen later.) If $T = \emptyset$, put $f_T(\mathbf{X}, \mathbf{Y}) \equiv 0$, while if $\emptyset \ne T \subseteq S$, where S is one of the blocks of the partition (in which case S is uniquely determined by T), define $f_T(\mathbf{X}, \mathbf{Y})$ by setting

$$f_T(x, y) = \begin{cases} 1, & \text{if } \{z \in S : f(x, y, z) = 1\} = T, \\ 0, & \text{otherwise.} \end{cases}$$

Observe that

$$f(\mathbf{X}, \mathbf{Y}, \mathbf{Z}) \equiv \bigvee_S \bigvee_{T \subseteq S} \left(f_T(\mathbf{X}, \mathbf{Y}) \wedge \bigvee_{z \in T} \mathbf{Z}^z \right). \tag{23}$$

Fix S. For each y, $T \subseteq S$, let $R(y, T) = \{x \in \mathcal{X} : f_T(x, y) = 1\}$. Define a function $f'_T(\mathbf{X}, \mathbf{Y})$ satisfying

$$f'_T(x, \mathbf{Y}) \equiv f_T(x, \mathbf{Y}), \quad \text{for all } x \in \mathcal{X}, \tag{24}$$

by requiring that for each y,

$$f'_T(\mathbf{X}, y) \equiv \bigvee_{x \in R(y,T)} \delta_x(\mathbf{X}),$$

where $\delta_x(\mathbf{X})$ is a conjunction of r literals satisfying (22).

Then $\mathbf{L}(f'_T(\mathbf{X}, y)) \le |R(y, T)|\, r$, so

$$\sum_{T \subseteq S} \mathbf{L}(f'_T(\mathbf{X}, y)) \le r\, |\mathcal{X}|,$$

since the sets $R(y, T)$ are disjoint, and $\bigcup_{T \subseteq S} R(y, T) \subseteq \mathcal{X}$. Therefore,

$$\sum_{T \subseteq S} \sum_y \mathbf{L}(f'_T(\mathbf{X}, y)) \le r\, |\mathcal{X}|\, 2^{|\mathbf{Y}|},$$

so by the Combining Lemma (Lemma 7.1),

$$\sum_{T \subseteq S} \mathbf{L}(f'_T(\mathbf{X}, \mathbf{Y})) \le r\, |\mathcal{X}|\, 2^{|\mathbf{Y}|} + 2^{|S|} . 2^{|\mathbf{Y}|+1}.$$

But by (23) and (24), for all $x \in \mathcal{X}$,

$$f(x, \mathbf{Y}, \mathbf{Z}) \equiv \bigvee_S \bigvee_{T \subseteq S} \left(f'_T(x, \mathbf{Y}) \wedge \bigvee_{z \in T} \mathbf{Z}^z \right).$$

Take $f'(\mathbf{X},\mathbf{Y},\mathbf{Z})$ to be the function defined for all x (not just $x \in \mathcal{X}$), by the expression on the right hand side. Then, remembering that $|S| \leq s$,

$$\begin{aligned} \mathbf{L}(f') &\leq \sum_S \left(\sum_{T \subseteq S} \mathbf{L}(f'_T) \;+\; \sum_{T \subseteq S} |\mathbf{Z}|\,|T| \right) \\ &\leq \left(\frac{2^{|\mathbf{Z}|}}{s} + 1 \right) \left(r\,|\mathcal{X}|\,2^{|\mathbf{Y}|} + 2^s\,2^{|\mathbf{Y}|+1} + 2^s\,|\mathbf{Z}|\,s \right) . \end{aligned}$$

We now choose s and $|\mathbf{Z}|$ so that the term $r\,|\mathcal{X}|\,2^{|\mathbf{Y}|}$ dominates sufficiently. Let $s = [\log|\mathcal{X}| - \log\log n] + 1$, $|\mathbf{Z}| = 2[\log r + \log\log n]$. Then $|\mathbf{Y}| \geq |\mathbf{Z}|$ (by the assumed lower bound on $|\mathbf{Y}|+|\mathbf{Z}|$). Also since $\mathcal{X}$ is r-special, it is obvious that $|\mathcal{X}| \leq 2^r\binom{n}{r}$, so $\log|\mathcal{X}| = \mathrm{O}(r\log n)$, and therefore $s = \mathrm{O}(r\log n)$. Consequently,

$$|\mathbf{Z}|\,s \;=\; \mathrm{O}(|\mathbf{Z}|\,r\log n) \;\leq\; \mathrm{O}(r^2\log^2 n) \;=\; \mathrm{O}(2^{|\mathbf{Z}|}) \;\leq\; \mathrm{O}(2^{|\mathbf{Y}|}),$$

so

$$\begin{aligned} r\,|\mathcal{X}|\,2^{|\mathbf{Y}|} + 2^s\,2^{|\mathbf{Y}|+1} + 2^s\,|\mathbf{Z}|\,s &= 2^{|\mathbf{Y}|}\,(\,r\,|\mathcal{X}| + \mathrm{O}(2^s)\,) \\ &= r\,|\mathcal{X}|\,2^{|\mathbf{Y}|}\left(1 + \mathrm{O}\left(\frac{1}{r\log n}\right)\right) . \end{aligned}$$

Also,

$$\frac{2^{|\mathbf{Z}|}}{s} + 1 \;\leq\; \frac{2^{|\mathbf{Z}|}}{\log|\mathcal{X}| - \log\log n}\left(1 + \mathrm{O}\left(\frac{r\log n}{r^2\log^2 n}\right)\right) ,$$

and the lemma follows.

■

All that is needed to complete a proof that $\mathbf{L}_{\max}(n) < (1+o(1))2^n/\log n$ is a family of r-special sets $\mathcal{X}_h$, $h \in H$, with the property that almost all (with a prescribed error term) assignments x lie in exactly one $\mathcal{X}_h$. For letting $\mathcal{X}_h(\mathbf{X})$ denote the characteristic function of $\mathcal{X}_h$, we can represent any function $f(\mathbf{X},\mathbf{Y})$ as

$$f(\mathbf{X},\mathbf{Y}) \;\equiv\; \bigvee_{h\in H} \left(f'_h(\mathbf{X},\mathbf{Y}) \wedge \mathcal{X}_h(\mathbf{X})\right) \;\vee\; \bigvee_{x \notin \bigcup_h \mathcal{X}_h} \left(f(x,\mathbf{Y}) \wedge \mathbf{X}^x\right),$$

where f'_h is obtained by applying Lemma 9.1 using $\mathcal{X}_h$. Clearly,

$$\mathbf{L}(f) \;\leq\; \sum_{h\in H}\left(\mathbf{L}(f'_h) + \mathbf{L}(\mathcal{X}_h)\right) \;+\; \sum_{x\notin\bigcup_h \mathcal{X}_h}\left(\mathbf{L}(f(x,\mathbf{Y})) + |\mathbf{X}|\right). \qquad (25)$$

The technique is to show that, for a suitable choice of $|\mathbf{X}|$, this is dominated by the upper bound on $\sum_{h\in H} \mathbf{L}(f'_h)$ obtained from Lemma 9.1. Actually in the applications given here we will be able to choose $\mathcal{X}_h$, $h \in H$ to be a partition of *all* the assignments x, in which case these simplify to:

$$\begin{aligned} f(\mathbf{X},\mathbf{Y}) &\equiv \bigvee_{h\in H} \Big(f'_h(\mathbf{X},\mathbf{Y}) \wedge \mathcal{X}_h(\mathbf{X}) \Big) , \\ \mathbf{L}(f) &\leq \sum_{h\in H} \Big(\ \mathbf{L}(f'_h) \ + \ \mathbf{L}(\mathcal{X}_h) \Big) . \end{aligned} \tag{26}$$

An example of a set which is obviously 1-special is the *surface* $\mathcal{X}_h$ of the $|\mathbf{X}|$-dimensional *sphere* of radius 1, centre h. That is,

$$\mathcal{X}_h = \{ x : \ x \text{ differs from } h \text{ in exactly 1 digit} \},$$

where both x and h have length $|\mathbf{X}|$. Clearly $\mathcal{X}_h$ has cardinality $|\mathcal{X}_h| = |\mathbf{X}|$.

If $|\mathbf{X}| = 2^m - 1$, for some integer m, then it is well known (now) that there exist sets H', the *Hamming codes*, consisting of assignments h' of length $|\mathbf{X}|$, such that every x lies in exactly one of the sets H'; $\mathcal{X}_{h'}, h' \in H'$. (Put in the language of coding theory, Hamming codes are perfect codes with minimum distance 3.) We could use this fact, which was relatively new when Lupanov first proved his theorem, together with (25). But it is neater (again following Lupanov) to take $|\mathbf{X}| = 2^m$, and let

$$H \ = \ \{ 0, h' : \ h' \in H' \} \cup \{ 1, h' : \ h' \in H' \} .$$

The 2^m dimensional spherical surfaces $\mathcal{X}_h$, $h \in H$ then obviously partition the set of *all* assignments x of length 2^m. (H is actually the union of the *extended Hamming code*, with one of its cosets. One explicit way to define H is:

$$H \ = \ \{ x : \sum_{\mathbf{e}\in\{0,1\}^m} x_e \mathbf{e} = \mathbf{0} \} ,$$

where $x = x_0, x_1, \ldots, x_{2^m-1}$ and we associate the subscript e of x_e with its binary representation vector $\mathbf{e} \in \{0,1\}^m$.)

COROLLARY 9.2 *(Lupanov's Theorem.) For every Boolean function f of n variables,*

$$\mathbf{L}(f) \ \leq \ \frac{2^n}{\log n} \Big(1 + \mathrm{O}(\log\log n / \log n) \Big) .$$

PROOF: Consider $f(\mathbf{X},\mathbf{Y})$, where $|\mathbf{X}|+|\mathbf{Y}|=n$ and $|\mathbf{X}|=2^{\lceil\log n\rceil-1}$. Notice that $|\mathbf{Y}|>|\mathbf{X}|$, and $\log|\mathbf{X}|=\log n-\mathrm{O}(1)$.

Take H to be the extended Hamming code of length $|\mathbf{X}|$ and define $\mathcal{X}_h$ as above. In order to apply (26), we need a bound on $\mathbf{L}(\mathcal{X}_h)$. Now

$$\mathcal{X}_h(\mathbf{X}) \;\equiv\; \bigvee_{x\in\mathcal{X}_h} \mathbf{X}^x,$$

which leads to the crude bound

$$\mathbf{L}(\mathcal{X}_h) \;\le\; |\mathcal{X}_h|\,|\mathbf{X}| \;=\; |\mathbf{X}|^2.$$

It follows by (26), Lemma 9.1 with $r=1$, and the equality $|H|\,|\mathbf{X}|=2^{|\mathbf{X}|}$, that

$$\begin{aligned}
\mathbf{L}(f) \;&\le\; \sum_{h\in H}\left(\frac{|\mathcal{X}_h|\,2^{|\mathbf{Y}|}\big(1+\mathrm{O}(1/\log n)\big)}{\log|\mathcal{X}_h|-\log\log n}+|\mathbf{X}|^2\right)\\
&\le\; |H|\,\frac{|\mathbf{X}|\;2^{|\mathbf{Y}|}\;\big(1+\mathrm{O}(1/\log n)\big)}{\log|\mathbf{X}|-\log\log n} \;=\; \frac{2^{|\mathbf{X}|}\,2^{|\mathbf{Y}|}\big(1+\mathrm{O}(1/\log n)\big)}{\log n-\log\log n-\mathrm{O}(1)}\\
&=\; \frac{2^n}{\log n-\log\log n-\mathrm{O}(1)}\,.
\end{aligned}$$

∎

It turns out that a better bound on the error term in Lupanov's Theorem can be obtained by considering r-special sets $\mathcal{X}_h$ with r about $\log\log n$.

One potential source of a family of such sets, consists of spherical surfaces of radius r around the elements of some code with minimum distance $2r+1$, which is "almost" perfect in that it comes close to meeting the "sphere packing bound". However there is no need for $\mathcal{X}_h$ to be a spherical surface. We will instead take $\mathcal{X}_h$ to be a Cartesian product of r 1-special sets, which will obviously be r-special.

Let $|\mathbf{X}|$ be of the form $|\mathbf{X}|=r\,2^m$ for some integer m. The sequence of variables $\mathbf{X}$ will be subdivided into r subsequences $\mathbf{X}_1,\ldots,\mathbf{X}_r$ of equal length $|\mathbf{X}_i|=2^m$. Any assignment x of length $|\mathbf{X}|$ can be similarly subdivided into assignments $x_1,\ldots,x_r$, each of length 2^m. Let $h=h_1,\ldots,h_r$ be some such assignment. Take

$$\mathcal{X}_h \;=\; \mathcal{X}_{h_1}\mathcal{X}_{h_2}\ldots\mathcal{X}_{h_r} \;=\; \{x_1,x_2,\ldots,x_r:\; x_i\in\mathcal{X}_{h_i}\;\text{ for } i=1,2,\ldots,r\}\,,$$

where

$$\mathcal{X}_{h_i} \;=\; \{x_i : \; x_i \text{ differs from } h_i \text{ in exactly 1 digit}\}.$$

Let $H^r = \{h_1, h_2, \ldots, h_r : \; h_i \in H \text{ for } i = 1, 2, \ldots, r\}$, where H is the extended Hamming code of length $|\mathbf{X}_i| = 2^m$. Clearly $\mathcal{X}_h$, $h \in H^r$ is a family of disjoint r-special sets.

THEOREM 9.3 *There is a constant $\kappa < 2$ such that for n sufficiently large,*

$$\mathbf{L}_{\max}(n) < \frac{2^n}{\log n - \log\log\log n - \kappa}\,.$$

Equivalently,

$$\log b(n, \mathbf{L}_{\max}(n)) > \log n - \log\log\log n - \kappa\,.$$

PROOF: Consider a function $f(\mathbf{X}, \mathbf{Y})$ of $n = |\mathbf{X}| + |\mathbf{Y}|$ variables. We will choose $|\mathbf{X}|$ in such a way that we can write $\mathbf{X} = \mathbf{X}_1, \ldots, \mathbf{X}_r$, where

$$|\mathbf{X}_i| \;=\; \frac{|\mathbf{X}|}{r} \;=\; 2^{\lceil \log n - \log\log\log n + \theta \rceil}\,, \tag{27}$$

and r is chosen to satisfy

$$(r+1)\,|\mathbf{X}_i| \;\le\; n \;<\; (r+2)\,|\mathbf{X}_i|\,. \tag{28}$$

Here θ is any constant in the interval $0 < \theta < 1$. (The reader may care to choose θ so as to optimise κ.)

From (27), $\log|\mathbf{X}_i| > \log n - \log\log\log n + \theta - 1$. Also $|\mathbf{X}_i| \le 2^\theta n / \log\log n$, so by (28), $r + 2 > 2^{-\theta}\log\log n$. Therefore

$$\begin{aligned}
\log|\mathbf{X}_i| - \frac{\log\log n}{r} \;&\ge\; \log n - \log\log\log n + \theta - 1 - 2^\theta - \mathrm{O}\left(\frac{1}{\log\log n}\right) \\
&>\; \log n - \log\log\log n - \kappa\,, \qquad (29)
\end{aligned}$$

for some constant $\kappa < 2$. Since $\log r \sim \log\log\log n$, (28) also ensures that

$$|\mathbf{Y}| \;=\; n - r\,|\mathbf{X}_i| \;\ge\; |\mathbf{X}_i| \;>\; 4(\log r + \log\log n)\,,$$

which will enable us to apply Lemma 9.1 to $f(\mathbf{X}, \mathbf{Y})$.

As in the proof of Corollary 9.2, we will use (26) (with H^r in place of H). Observing that

$$\mathcal{X}_h(\mathbf{X}) \;\equiv\; \bigwedge_{i=1}^{r} \mathcal{X}_{h_i}(\mathbf{X}_i)\,,$$

the crude bound $\mathbf{L}(\mathcal{X}_{h_i}(\mathbf{X}_i)) \leq |\mathbf{X}_i|^2$ implies that

$$\mathbf{L}(\mathcal{X}_h(\mathbf{X})) \leq r\,|\mathbf{X}_i|^2 = |\mathbf{X}|^2/r\,,$$

so (26) and Lemma 9.1 yield:

$$\begin{aligned}
\mathbf{L}(f) \;&\leq\; \sum_{h\in H^r} \left(\frac{r\,|\mathcal{X}_h|\,2^{|\mathbf{Y}|}\left(1+\mathrm{O}(1/(r\,\log n))\right)}{\log|\mathcal{X}_h| - \log\log n} + \frac{|\mathbf{X}|^2}{r} \right) \\
&\leq\; \frac{r\,2^{|\mathbf{X}|}\,2^{|\mathbf{Y}|}\left(1+\mathrm{O}(1/(r\,\log n))\right)}{r\,\log|\mathbf{X}_i| - \log\log n} \qquad (30)
\end{aligned}$$

Here we have used $|\mathcal{X}_h| = |\mathcal{X}_{h_i}|^r = |\mathbf{X}_i|^r$, and the fact that $\sum_{h\in H^r} |\mathcal{X}_h| = 2^{|\mathbf{X}|}$ because the sets $\mathcal{X}_h$ are a partition. Using (29), it follows immediately that

$$\mathbf{L}_{\max}(n) \;\leq\; \frac{2^n}{\log n - \log\log\log n - \kappa - \mathrm{O}(1/\log\log n)}\,.$$

■

The following general lemma is designed for the situation where we have a lower bound on $b(n, l(n))$ (for example with $l(n) = \mathbf{L}_{\max}(n)$) of the form

$$\log b(n, l(n)) \;\geq\; h(n) - o(g(n))\,,$$

so typically $h(n) \sim \log n$. It says that if $h(n), g(n)$ are "nice" functions, then a similar bound on $b(n, L)$ will hold for a range of values of L smaller than $l(n)$.

Lemma 9.4 *Consider positive real valued functions $g(n) \geq 1$, $h(n) \geq g(n)$, and positive integer valued functions $m(n), l(n), l_1(n)$. Writing $m = m(n)$, assume that as $n \to \infty$, these functions have the properties:*

$$m < n\,, \qquad m \to \infty\,, \qquad |h(n) - h(m)| = o(g(n))\,,$$

$$g(m) \sim g(n)\,, \qquad \frac{l_1(n)}{2^m} \to \infty\,, \qquad \frac{l(m)}{n\,h(n)} \to \infty\,.$$

Suppose also that $\log b(n, l(n)) \geq h(n) - o(g(n))$. *Then*

$$\log b(n, L) \;\geq\; h(n) - o(g(n))\,, \tag{31}$$

uniformly for all L *satisfying* $l_1(n) \leq L \leq 2^{n-m} l(m)$. *Moreover, for the function* $l(n) = \mathbf{L}_{\max}(n)$, (31) *holds uniformly for* $l_1(n) \leq L \leq \mathbf{L}_{\max}(n)$.

PROOF: It suffices to show that for every choice of constants ϵ,ϵ', with $0 < \epsilon < \epsilon' < 1$, if $\log b(n, l(n)) \geq h(n) - \epsilon g(n)$ eventually, then for all L in the specified range, $\log b(n, L) \geq h(n) - \epsilon' g(n)$, whenever n is sufficiently large. (How large depends only on ϵ', not on L.)

Split the n variables into two disjoint subsequences $\mathbf{X}, \mathbf{Y}$ with $|\mathbf{X}| = m$, $|\mathbf{Y}| = n - m$. Let $\mathcal{Y} \subseteq \{0,1\}^{|\mathbf{Y}|}$. ($|\mathcal{Y}|$ will be chosen later.) Consider those Boolean functions $f(\mathbf{X}, \mathbf{Y})$ which have $f(\mathbf{X}, y) \equiv 0$ for all $y \notin \mathcal{Y}$, and $\mathbf{L}(f(\mathbf{X}, y)) \leq l(|\mathbf{X}|)$ for all $y \in \mathcal{Y}$. As

$$f(\mathbf{X}, \mathbf{Y}) \;\equiv\; \bigvee_{y \in \mathcal{Y}} \Big(f(\mathbf{X}, y) \;\wedge\; \mathbf{Y}^y \Big)\,,$$

it follows immediately that

$$\mathbf{L}(f) \;\leq\; |\mathcal{Y}|\,(\,l(|\mathbf{X}|) + |\mathbf{Y}|\,) \;\leq\; |\mathcal{Y}|\,(l(m) + n)\,.$$

In order to find the conditions under which $\mathbf{L}(f) \leq L$, it is convenient to prove that the estimate

$$l(m) + n \;\leq\; \frac{\log B(m, l(m))}{h(n) - \epsilon'' g(n)}\,,$$

is ultimately satisfied, when ϵ'' is any constant with $\epsilon < \epsilon'' < \epsilon'$. To see this, recall that $\log B(m, l(m)) = l(m) \log b(m, l(m))$ and, by assumption, $\log b(m, l(m)) \;\geq\; h(m) - \epsilon g(m) \;=\; h(n) - (\epsilon + o(1))\, g(n)$. Hence,

$$\begin{aligned}
\frac{\log B(m, l(m))}{h(n) - \epsilon'' g(n)} \;&\geq\; l(m)\, \frac{h(n) - (\epsilon + o(1))\, g(n)}{h(n) - \epsilon'' g(n)} \\
&=\; l(m) \left(1 + \frac{(\epsilon'' - \epsilon - o(1))\, g(n)}{h(n) - \epsilon'' g(n)} \right)\,.
\end{aligned}$$

Moreover, using the assumptions $h(n) \geq g(n) \geq 1$ and $l(m)/(n\, h(n)) \to \infty$, we have for every large enough n, that

$$l(m)\, \frac{(\epsilon'' - \epsilon - o(1)) g(n)}{h(n) - \epsilon'' g(n)} \;\geq\; l(m)\, \frac{\epsilon'' - \epsilon - o(1)}{h(n) - \epsilon''} \;\geq\; n\,.$$

Combining this with the previous bound, gives the required estimate.

Therefore to ensure $\mathbf{L}(f) \leq L$, it suffices to choose $|\mathcal{Y}|$ such that

$$|\mathcal{Y}| \log B \;\leq\; (h(n) - \epsilon'' g(n))\, L\,, \tag{32}$$

where $B = B(m, l(m))$.

Considering only functions f of the above form, we see that

$$B(n, L) \;\geq\; B(|\mathbf{X}|, l(|\mathbf{X}|))^{|\mathcal{Y}|} \;=\; B^{|\mathcal{Y}|},$$

so $\log b(n, L) \geq h(n) - \epsilon' g(n)$ will hold, if in the choice of $|\mathcal{Y}|$, we can also satisfy

$$|\mathcal{Y}| \log B \;\geq\; (h(n) - \epsilon' g(n))\, L\,. \tag{33}$$

In the *interval* of $|\mathcal{Y}|$ values defined by (32) and (33), there will be some *integer* $|\mathcal{Y}|$ with $1 \leq |\mathcal{Y}| \leq 2^{|\mathbf{Y}|}$, if

$$(\epsilon' - \epsilon'')\, g(n)\, L \;\geq\; \log B\,,$$

and

$$\frac{(h(n) - \epsilon'' g(n))\, L}{\log B} \;\leq\; 2^{|\mathbf{Y}|} \;=\; 2^{n-m}\,.$$

The first of these holds as

$$(\epsilon' - \epsilon'')\, g(n)\, L \;\geq\; (\epsilon' - \epsilon'')\, L \;\geq\; 2^m \;=\; \log B(m, \mathbf{L}_{\max}(m)),$$

using the assumptions $L/2^m \geq l_1(n)/2^m \to \infty$. The second holds because

$$L \;\leq\; 2^{n-m} l(m) \;\leq\; \frac{2^{n-m} \log B}{h(m) - \epsilon g(m)} \;\leq\; \frac{2^{n-m} \log B}{h(n) - \epsilon'' g(n)}\,,$$

or in the case $l(n) = \mathbf{L}_{\max}(n)$, because

$$L \;\leq\; \mathbf{L}_{\max}(n) = l(n) = \frac{2^n}{\log b(n, l(n))} \;\leq\; \frac{2^n}{h(n) - \epsilon'' g(n)} \;=\; \frac{2^{n-m} \log B}{h(n) - \epsilon'' g(n)}\,.$$

■

In all three applications of this lemma given here, we will take $g(n) = 1$. (An application with non-constant $g(n)$ is suggested at the end of Section 2.) We will state the theorems using n as a lower bound for L, but in fact any function going to infinity with n will do.

THEOREM 9.5 *Suppose* $b(n, \mathbf{L}_{\max}(n)) \geq (c - o(1))n$. *Then*

$$b(n, L) \sim cn$$

uniformly for $n \leq L \leq \mathbf{L}_{\max}(n)$.

PROOF: By Theorems 6.2 and 8.3, it suffices to show that $\log b(n, L) \geq \log cn - o(1)$ uniformly for $L \geq l_1(n)$, where $\log l_1(n) = n - [\log n]^3$. Take $h(n) = \log cn$. Then for $m < n$,

$$h(n) - h(m) = \log \frac{n}{m} = -\log\left(1 - \frac{n-m}{n}\right) \sim \frac{n-m}{n} = o(1)$$

provided $m = n - o(n)$. We can certainly choose m satisfying this condition and also having $\log l_1(n) - m \to \infty$. So the conditions of the lemma are all satisfied (with $g(n) = 1$, $l(n) = \mathbf{L}_{\max}(n)$).

■

THEOREM 9.6 *Suppose there is some function* $l(n) \sim \mathbf{L}_{\max}(n) \sim 2^n / \log n$ *such that* $b(n, l(n)) \geq (c - o(1))n$. *Then there is some function* $l_*(n) \sim \mathbf{L}_{\max}(n)$ *such that*

$$b(n, L) \sim cn$$

uniformly for $n \leq L \leq l_*(n)$.

PROOF: Choosing $m = m(n)$ similarly to the last proof, we see that the conclusion holds for

$$n \leq L \leq 2^{n-m} l(m) \sim 2^{n-m} \frac{2^m}{\log m} \sim \frac{2^n}{\log n},$$

so $l_*(n)$ can be taken to be $2^{n-m(n)} l(m(n))$.

■

Notice that if we replace c by a constant $c' < c$ in the assumptions of Theorems 9.5 and 9.6, the argument still allows us to conclude that

$b(n, L) \geq (c' - o(1))n$. But even in this weakened form, we do not know whether either assumption holds. In particular, we do not know whether for all large n, $b(n, \mathbf{L}_{\max}) > n$, or equivalently, whether $\mathbf{L}_{\max}(n) < 2^n / \log n$. However Theorem 9.3 does tell us that

$$\log b(n, \mathbf{L}_{\max}(n)) > \log n - \log\log\log n - \kappa$$

for some constant $\kappa < 2$.

THEOREM 9.7 *If n is sufficiently large, then*

$$b(n, L) \ > \ \frac{n}{4 \log\log n}$$

for all L satisfying $n \leq L \leq \mathbf{L}_{\max}(n)$.

PROOF: Take $h(n) = \log n - \log\log\log n - \kappa$ and modify the proof of Theorem 9.5 in the obvious way, to conclude that

$$\log b(n, L) \ \geq \ \log n - \log\log\log n - \kappa - o(1) \ > \ \log n - \log\log\log n - 2\,.$$

■

For all values of $L \leq \mathbf{L}_{\max}(n)$, it would definitely be preferable to have an asymptotic formula for $B(n, L)$, rather than just our formula (and error term estimates) for $\log B(n, L)$. A successful extension of the methods used in [11] and [31], where n is held fixed, and $L \to \infty$, to the case where both L, n go to infinity in a fixed relationship, might help illuminate this problem.

Acknowledgements.
The second author would like to thank the Institute of Computer Science of the Czech Academy of Sciences for accommodating him, and both the Institute of Computer Science, and the Institute of Mathematics, together with their members, for their hospitality throughout his visit to Prague, during which initial work on this topic was done. Thanks also, to the numerous other benefactors who made the trip possible.

References

[1] D. Angluin, L. Hellerstein and M. Karpinski. Learning read-once formulas with queries. *J. Assoc. Comput. Mach.*, 40:185–210, 1993.

[2] A. Cayley. On the theory of the analytical forms called trees. *Phil. Mag.*, 13:172–176, 1857.

[3] L. Comtet. Sur le quatrième problème et les nombres de Schröder. *C. R. Acad. Sci. Paris Sér. A Math.*, 271:913–916, 1970.

[4] P. E. Dunne. *The Complexity of Boolean Networks.* Academic Press, London, 1988.

[5] T. Eiter. Generating μ-expressions. *Acta Inform.*, 32:171–187, 1995.

[6] L. R. Foulds and R. W. Robinson. Enumeration of phylogenetic trees without points of degree two. *Ars Combin.*, 17A:169–183, 1984.

[7] J. Friedman. Probabilistic spaces of Boolean functions of a given complexity: generalities and random k-SAT coefficients. Research Report CS-TR-387-92, Princeton University, Princeton, NJ, 1992.

[8] F. Harary and E. M. Palmer. *Graphical Enumeration.* Academic Press, New York, 1973.

[9] F. Harary, R. W. Robinson and A. J. Schwenk. Twenty–step algorithm for determining the asymptotic number of trees of various species. *J. Austral. Math. Soc., Series A*, 20:483–503, 1975.

[10] H. B. Hunt and R. E. Stearns. The complexity of very simple Boolean formulas with applications. *SIAM J. Comput.*, 19:44–70, 1990.

[11] H. Lefmann and P. Savický. Some typical properties of large AND/OR Boolean formulas. In *Proceedings MFCS'95, Lecture Notes in Computer Science 969*, pages 237–246, 1995.

[12] O. B. Lupanov. Complexity of formula realization of functions of logical algebra. *Problemy Kibernetiki*, 3:61–80, 1960. English translation, *Problems of Cybernetics*, 3:782–811, 1962, Pergamon Press.

[13] O. B. Lupanov. On the realization of functions of logical algebra by formulae of finite classes (formulae of limited depth) in the basis $\&, \vee, ^{-}$. *Problemy Kibernetiki*, 6:5–14, 1961. English translation, *Problems of Cybernetics*, 6:1–14, 1965, Pergamon Press.

[14] P. A. MacMahon. The combination of resistances. *The Electrician*, 28:601–602, 1892.

[15] A. I. Markushevich. *Theory of Functions of a Complex Variable, Volume II.* Prentice-Hall, 1965.

[16] J. W. Moon. Some enumerative results on series-parallel networks. In *Random Graphs '85*, pages 199–226. New Holland, Amsterdam, 1987.

[17] D. Mundici. Solution of Rota's problem on the order of series-parallel networks. *Adv. in Appl. Math.*, 12:455–463, 1991.

[18] R. Otter. The number of trees. *Ann. of Math.*, 49:583–599, 1948.

[19] J. B. Paris, A. Vencovská, and G. M. Wilmers. A natural prior probability distribution derived from the propositional calculus. *Ann. Pure Appl. Logic*, 70:243–285, 1994.

[20] G. Pólya. Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Math.*, 68:145–254, 1937.

[21] G. Pólya and R. C. Read. *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds.* Springer-Verlag, New York, 1987.

[22] J. Riordan. *An Introduction to Combinatorial Analysis.* Wiley, New York, 1958.

[23] J. Riordan and C. E. Shannon. The number of two terminal series-parallel networks. *Journal of Math. and Physics*, 21:83–93, 1942.

[24] J. E. Savage. *The Complexity of Computing.* Wiley, 1976.

[25] P. Savicky. Complexity and probability of some Boolean formulas. Preprint, 1996.

[26] E. Schröder. Vier combinatorische Probleme. *Z. Math. und Physik*, 15:361–376, 1870.

[27] R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms*. Addison Wesley Publishing Company, 1996.

[28] C. E. Shannon. The synthesis of two-terminal switching circuits. *Bell System Technical Journal*, 28:59–98, 1949.

[29] I. Wegener. *The Complexity of Boolean Functions*. Wiley–Teubner, Stuttgart, 1987.

[30] H. S. Wilf. *generatingfunctionology*. Academic Press, Boston, 1990.

[31] A. R. Woods. Coloring rules for finite trees, and probabilities of monadic second order sentences. *Random Struct. Alg.*, 10:453–485, 1997.

Institute of Computer Science
Academy of Sciences of the Czech Republic
Prague
CZECH REPUBLIC

Email: savicky@uivt.cas.cz

Department of Mathematics
University of Western Australia
Nedlands W.A. 6907
AUSTRALIA

Email: woods@maths.uwa.edu.au