# Quantitative comparison of languages

András Kornai (kornai@bbn.com)
*BBN Technologies*
*70 Fawcett Street, Cambridge, MA 02138*

**Abstract.** From the perspective of the linguist, the theory of formal languages serves as an abstract model to address issues such as complexity, learnability, information content, etc. which are hard to investigate directly on natural languages. One question that has not been sufficiently addressed in the literature is to what extent can a result proved on an abstract model be presumed to hold for the concrete languages that are, after all, the real object of interest in linguistics. In this paper we attempt to remedy this defect by developing some figures of merit that measure how well a formal language approximates an actual language. We will review and refine some standard notions of mathematical *density* to arrive at a numerical figure that shows the degree to which one language approximates another, and show how such a figure can be computed between some formal languages and empirically measured between a real language and its formal model. In the concluding section of the paper we will argue that from the statistical perspective developed here even some classical results of mathematical linguistics, such as Chomsky's (1957) demonstration of the inadequacy of finite state models, are highly suspect.

**Keywords:** density, regular languages, complexity

## 0. Introduction

Imagine the issue is some dimensionless physical constant in nature. Like any other number, this constant can be rational, algebraic, or transcendental. When someone measures the number and finds it to be 3.141592 with the six decimals of precision afforded by the measurement, she is entitled to say the number is, within the error of measurement, close to $\pi$. She can then go on to prove, with full mathematical rigor, that $\pi$ is transcendental. However, she is not permitted to jump to the conclusion that the original physical constant was transcendental, for it could just as well be approximated by a rational such as 355/113.

Remarkably, a similarly fallacious argument has been widely accepted in linguistics for forty years. We are referring, of course, to Chomsky's (1957) demonstration that natural languages are not regular. As we shall see in the concluding sections of this paper, this demonstration rests on the impermissible step of substituting a mathematical model, formal languages, for the actual object of inquiry, which are natural languages. What makes the physical constant example trivial is that the relationship between a number and other

© 2006 *Kluwer Academic Publishers. Printed in the Netherlands.*

numbers *approximating* it is well understood. In this paper we discuss how Language Models (a weighted generalization of formal languages) can approximate Languages (a subclass of weighted formal languages) as well as formal or natural languages. The key notion of this approximation is *density,* defined in Section 1 and studied in some detail for the regular case in Section 2. Finer measures of approximation for zero density Languages are discussed in Section 3, and are related to standard notions like *channel capacity* in special cases.

After these preparations, in Section 4. we turn to Chomsky's self-embedding argument and show that it remains fallacious even if we accept the competence vs. performance distinction familiar from the early years of the debate surrounding the subject. Finally, in Section 5. we discuss the related "Infinity Fallacy" that no significant generalization about the syntax of a language can rest on a finite class of cases. Mathematical arguments that would stretch the boundaries of this paper are relegated to the Appendix.

It is impossible for a paper about such a controversial matter to be entirely free of polemic. But the reader is exhorted to focus on the positive contributions of the article, which include a conservative extension of the traditional notion of density, a characterization of zero density regular languages, and a simple yet powerful definition of approximation error, and take the polemic in the spirit it is offered, with the goal of clarifying a complex issue, rather than saying the final word.

## 1. Definitions

Our central notion will be that of a *Language* (with capital L) over a finite alphabet of phonemes or graphemes $V$ (including pause or whitespace) which we define as a function $f$ that assigns a non-negative probability $f(\alpha)$ to every string $\alpha$ over $V$ in such a manner that the sum of these is bounded (can be normalized to 1). To forestall confusion it should be emphasized that probability is not meant as a numerical scale of degrees of grammaticality: syntactically well-formed strings can have zero or low probability and syntactically ill-formed strings can have relatively high probability. The definition of Language embodies the simplifying assumption that string probabilities are fixed once and for all, though in estimation tasks it has been often noted that for low values the static probabilities are outweighed by context effects. To fix ideas, a Language is best thought of as the set of strings that will be encountered by an idealized speaker/hearer (or by a computer application such as a speech or character recognition device), weighted by the frequency of such encounters. For the most part we will ignore the fact that language changes with time and therefore such frequencies are not truly fixed once and for all – we will use a stationary model that assumes we can average over past, present, and future inputs. A *Language Model* is defined as any combination of table lookup and other algorithmic procedures that will assign a non-negative number to each string over the alphabet. We do not require the sum of these numbers to converge because we do not wish to exclude language models like formal languages which approximate probabilities by a 0-1 decision but permit an infinite number of valid strings.

Eilenberg 1974 (p 225) defines the density of a language $L$ over a one-letter alphabet as

$$\lim_{n\to\infty} \frac{|\{\alpha \in L : |\alpha| < n\}|}{n}$$

if this limit exists. This definition can be generalized for languages over a $k$-letter alphabet $V$ in a straightforward manner: if we arrange the elements of $V^*$ in a sequence $\phi$ and collect the first $n$ members of $\phi$ in the sets $V_n$,

$$\lim_{n\to\infty} \frac{|L \cap V_n|}{n} = \rho_\phi(L) \tag{1}$$

can be interpreted as the density of $L$ when it exists. Since this definition is not independent of the choice of the ordering $\phi$, we need to select a canonical ordering. We will call an ordering $\psi$ *length-compatible* if $|\psi(n)| \leq |\psi(m)|$ follows from $n < m$. It is easily seen that for arbitrary alphabet $V$ and language $L$, if $\phi$ is a length-compatible ordering of $V^*$ and the limit in (1) exists, than it exists and has the same value for any other length-compatible ordering $\psi$. In such cases we can in fact restrict attention to the subsequence of (1) given by $V^0, V^1, V^2, \ldots$: if we denote the number of strings of length $n$ in $L$ by $r_n$, *natural density* $\nu$ can be defined by

$$\nu(L) = \lim_{n\to\infty} \frac{\sum_{i=0}^{n} r_i}{\sum_{i=0}^{n} k^i} \tag{2}$$

This is in fact the definition used in Berstel (1973) and subsequent work. But to define density by (2) over $k$-letter alphabets for $k > 1$ has considerable drawbacks, since this limit fails to converge for some simple languages, such as the one containing all and only strings of even length. To avoid such problems, we introduce the generating function $d(z) = \sum_{n=0}^{\infty} r_n z^n$, and define *Abel density* $\rho$ by

$$\rho(L) = \lim_{z\to 1} (1 - z) d(z/k) \tag{3}$$

if this limit exists. A classical theorem of Hardy and Littlewood asserts that whenever the limit (2) exists (3) will also exist and have the same value, so our definition is conservative. We use the name Abel density because we replaced the Cesàro-summation implicit in Berstel's and Eilenberg's definition with Abel-summation. In the case of Languages, the number of strings $r_n$ are replaced by sums $R_n$ of the probabilities of the strings of length $n$, but otherwise the definition in (3) can be left unchanged. Even in Language Models where the weights do not necessarily sum to one, as long as every weight is maximum one, the Abel density will never be less than zero or more than one.

Beauquier and Thimonier (1986) define the *Bernoulli density* $\delta$ of a language $L$ through Language Models (though they do not use this term) that arise through a Bernoulli process in which each element $a_i$ of the alphabet has some fixed probability $p_i$ ($\sum_{i=1}^{k} p_i = 1$) and the weight $f(\alpha)$ of a string $\alpha$ is the product of the probabilities of its letters. Since the sum of all such weights would be divergent for any $L$, they consider only prefixes (minimal left factors) in $L$. Using $\lambda$ for the empty word, $\alpha \in L$ is a prefix iff for any decomposition $\alpha = \beta\gamma$, $\gamma = \lambda$ follows from $\beta \in L$. Denoting the prefixes in $L$ by $Pref(L)$, Bernoulli density is defined by

$$\delta(L) = \sum_{\alpha \in Pref(L)} f(\alpha) \tag{4}$$

In the equiprobable case, for languages where every word is a prefix, this coincides with the *combinatorial density* $\kappa(L) = \sum_{n=1}^{\infty} r_n/k^n$. It is easy to see that for $k > 1$ if a positive natural density $\nu$ exists the terms in $\kappa$ will converge to $\nu$ so combinatorial density itself will diverge.

While (3) does not always yield a numerical value (e.g. the properly context-sensitive language $\{a^i : 4^n \leq i < 2{\cdot}4^n, n \geq 0\}$ can be shown not to have Abel density over the one-letter alphabet $\{a\}$), Bernoulli density always exists. Though this suggests that Bernoulli density would be a better candidate for a basic measure in quantitative comparisons than the Abel density developed above, there is an important consideration that points in the other direction: Abel density, when it exists, is always additive (because of the absolute convergence of the power series in $z = 1$), while Bernoulli density is only an exterior measure, additive only for languages closed under right multiplication. If $L$ is not closed, there is an $\alpha \in L$ and a $\beta \in V^*$ such that $\alpha\beta \notin L$. Either $\alpha$ is a prefix or it contains a left factor $\alpha_0 \in L$ which is. Consider the two-member language $X = \{\alpha_0, \alpha\beta\}$:

$$\delta(X) = p(\alpha_0) \neq p(\alpha_0) + p(\alpha\beta) = \delta(X \cap L) + \delta(X \setminus L)$$

Thus, by Caratheodory's theorem, $L$ cannot be measurable. Note also that for languages closed under right multiplication $r_{n+1} \geq kr_n$ so the coefficients in $d(z/k) = \sum_{n=0}^{\infty} r_n z^n/k^n$ are non-decreasing. Therefore the coefficients of the Taylor expansion of $(1-z)d(z/k)$ are non-negative, and the Abel density $\rho$ also exists.

## 2. The regular case

As Berstel notes (p 346), neither combinatorial nor natural density will always exist for regular languages, even for relatively simple ones, such as the language of even length strings over a two-letter alphabet. Abel density does not suffer from this problem:

**Theorem 1.** Let $L$ be a regular language over some $k$-letter alphabet $V$. The Abel density $\rho(L)$ defined in (3) always exists, and is the same as the natural density whenever the latter exists. The Abel density of a regular language is always a rational number between 0 and 1.

The proof (see the Appendix) makes clear that shifting the initial state to $i'$ will mean only that we have to compute $\vec{v}H\vec{e}_{i'}$ with the same limiting matrix $H$, so density is a bilinear function of the (weighted) choice of initial and final states. The density vector $H\vec{e}_i$ can be easily computed if the graph of the finite deterministic automaton accepting L is strongly connected: in this case the Frobenius-Perron theorem can be applied to show that the eigenvalue $k$ of the transition matrix has multiplicity 1, and the density vector is simply the eigenvector corresponding to $k$ normed so that the sum of the components is 1. If this condition does not hold, the states of the automaton have to be partitioned into strongly connected equivalence classes: such a class is *final* if no other class can be reached from it, otherwise it is *transient.*

**Theorem 2.** The segment corresponding to a final class in the overall density vector is a scalar multiple of the density vector computed for the class in question. Those components of the density vector which correspond to states in some final class are strictly positive, and those which correspond to states in the transient class are 0.

We will say that a language $L$ over $V$ is *blocked* by a string $\beta$ if $L\beta V^* \cap L = \emptyset$. $L$ is *vulnerable* if it can be blocked by finitely many strings i.e. iff

$$\forall\alpha \in L \exists\beta \in \{\beta_1, \ldots, \beta_s\} \forall\gamma \in V^* \alpha\beta\gamma \notin L$$

**Theorem 3.** For a language $L$ accepted by some finite deterministic automaton $\mathcal{A}$ the following are equivalent:

(i) $\rho(L) = 0$

(ii) The accepting states of $\mathcal{A}$ are transient

(iii) $L$ is vulnerable.

Natural languages are vulnerable: it is easy to provide ungrammatical strings which, once appended to a left factor of a sentence, will make recovery impossible for any left factor that does not explicitly introduce a metalinguistic/quotation complement, and those left factors which do introduce such complements can be blocked by another string explicitly closing them and then introducing unrecoverable ungrammaticality.

Another important application of Theorem 3. is to the formal languages that can be generated/accepted by Hidden Markov Models. Such languages are not only regular but also locally testable, and therefore (except for $L = V^*$) can always be blocked by any string that does not appear as a local subword of some word in the language.

## 3. Finer distinctions

Since the most important Language Models have zero density, it is of great importance to introduce finer quantitative measures. One such measure could be the *saturation* $\sigma(L)$ of a language $L$ over a $k$-letter alphabet, which is given by the reciprocal of the convergence radius of $d(z/k)$.

**Theorem 4.** If $\rho(L) > 0$ then $\sigma(L) = 1$. If $\rho(L) = 0$ and $L$ is regular then $\sigma(L) < 1$. If $L \subset L'$ then $\sigma(L) \leq \sigma(L')$. $\sigma(L) = 0$ iff $L$ is finite.

Saturation has a further advantage: neither Bernoulli nor combinatorial density can be invariant under multiplication with an arbitrary string, but for Abel density and for saturation we have $\forall L, \alpha \rho(L) = \rho(\alpha L) = \rho(L\alpha), \sigma(L) = \sigma(\alpha L) = \sigma(L\alpha)$. Much as Abel density, saturation generalizes trivially from languages to Languages and Language Models.

**Example.** Let $D_1$ be the Dyck language over a two-letter alphabet. $r_{2n} = \binom{2n}{n}/(n+1)$, so $d(z) = (1 - \sqrt{1-4z^2})/2z^2$, so $d(z/2)$ will have its first singularity in 1. However, if $D_1$ is generated by the grammar $S \rightarrow aSb|SS|\lambda$ then the generating function associated with the grammar (Chomsky and Schützenberger 1963) will satisfy the functional equation $d(z) = z^2 d(z) + d^2(z) + 1$ and will therefore have its first singularity in $\sqrt{3}$. The Language Model where the weight of a string is given by the number of derivations it has is supersaturated: its saturatedness $2\sqrt{3}$ can be interpreted as the degree of its ambiguity.

If strings of length $n$ are generated by some CFG approximately $a_n$ times, then $a_{n+m} \approx a_n a_m$, because context-freeness makes disjoint subtrees in the generation tree independent. Therefore, $a_n \approx c^n$ and the base $c$ is a good measure of ambiguity. By the Cauchy-Hadamard theorem, $\sigma = \limsup \sqrt[n]{a_n} = c$. Note also that in the unambiguous case $\log(\sigma) = \limsup \log(a_n)/n$ can be interpreted as the *channel capacity* of the grammar (Kuich 1970). In the unambiguous case as well as in the case of context-free Languages where weights are given by the degree of ambiguity, the generating functions corresponding to the nonterminals satisfy a system of algebraic equations, and therefore $d(z)$ will have its first singularity in an algebraic point, therefore in such cases saturation and Abel density are algebraic numbers.

One problem with saturation is that it provides an all or nothing type decision when investigating whether one Language Model approximates another. For example, the language $D_1^1$ which permits matched parentheses of depth one, as given by the grammar $S \rightarrow aTb|SS|\lambda, T \rightarrow ab|abT$, can be subtracted from $D_1$ but the resulting Language Model has the same saturation as the original. Therefore we introduce two more direct measures, but unlike the case of natural density vs. Abel density, where the change is essentially technical, here we depart more significantly from the pioneering work of Berstel.

Given an alphabet $T$, a Language $f : T^* \rightarrow R^+$, a Language Model $g : T^* \rightarrow R^+$, and a precision $\epsilon > 0$, we define the *underestimation error* $U(\epsilon)$ of $g$ with respect to $f$ by

$$U(\epsilon) = \sum_{\substack{\alpha \in T^* \\ g(\alpha) < f(\alpha) - \epsilon}} f(\alpha) - g(\alpha) \tag{5}$$

and the *overestimation error* by

$$T(\epsilon) = \sum_{\substack{\alpha \in T^* \\ g(\alpha) > f(\alpha) + \epsilon}} g(\alpha) - f(\alpha) \tag{6}$$

While Berstel considers pairs of languages $L$ and mappings $f$ from $V^*$ to $R^+$ (these mappings are the same as our Language Models except for the fact that Berstel requires strictly positive values while in LMs zero values are also permitted), we are considering pairs of Language Models. Also, Berstel computes the ratio of the summed weights (summed for strings of length $\leq n$ in $L$ in the numerator, for all strings in $V^n$ in the denominator), while our definition uses the differences, segregated by sign. We believe that our choice reflects the practice of computational language modeling better, inasmuch as in practical Language Models both underestimation and overestimation errors are present, and their overall effects are seldom determined in the limit (incorrect estimates for high frequency strings are far more important than incorrect estimates for low frequency strings).

Of particular interest is the case $\epsilon = 0$ where all instances of underestimation and overestimation are taken into account. In the case of approximating the CFG we used above to generate $D_1$ by the CFG used above to define $D_1^1$ there is only underestimation error, which becomes convergent e.g if we assume that the negative log probability of a string is proportional to its length. If the constant of proportionality is chosen as $\log(k)$, combinatorial density is simply the overestimation error $T(0)$ of the Language Model with respect to the empty language.

## 4. Self-embedding

In general, the proportion of "interesting" strings among all combinatorially possible strings of the same length decreases rapidly, so the Abel density of interesting languages is always zero. Generalizing to Languages means that $d(1) = \sum_{i=0}^{\infty} R_i = 1$, so $\lim_{z\to 1}(1-z)d(z/k)$ will again be 0. In other words, only Language Models with a divergent weight sum can give rise to nonzero density. This effect is somewhat mitigated by Bernoulli density, which ignores every string that has a valid prefix. But when approximating one Language by another, concentrating only on prefixes is counterintuitive, and it makes sense to keep underestimation and overestimation errors separate.

From this perspective, several arguments advanced in mathematical linguistics about the structure and complexity of natural language stringsets are highly suspect: we will illustrate this on the classic argument of Chomsky (1957) against finite state models. While the standard counterargument (Yngve 1961) focuses on the breakdown between the competence and the performance of the speaker/hearer, here we fully accept the claim that self-embedding to any degree is grammatical. In fact, we permit our Language Model to contain several rules that will self-embed (either in themselves or in combination with other rules). But if our model makes 0-1 decisions (does not assign probabilities to strings) its overestimation error will always be infinite, since it generates an infinite number of strings by the self-embedding construction alone, while the total probability of such strings in natural language is less than one.

There are several ways to bring this divergence under control: normalizing factors (e.g. negative log probabilities proportional to the length of the string) can be applied, an arbitrary cutoff point in length can be selected, etc. Once this has been done, we can ask: how much would we increase underestimation error if self-embedding was limited in the model? While string frequencies are often hard to measure, it should be emphasized that there is no deep philosophical issue here, only the pragmatic issue of how to deploy limited resources optimally. It is easy to verify that simply embedded constructions take less than 3% of e.g. the Brown corpus, doubly embedded constructions less than 3% of that, and so on. Given that the under/overestimation errors of every current computational model are an order of magnitude higher (in the 30% range), an argument based on a data set containing at most $\sum_{n=1}^{\infty} .03^n$ (i.e. less than 3.1%) of the data is relegated to the status of a curiosity.

Most empirical fields of study have their share of curiosities, and it is hard to find a theoretical model that gives rise to absolutely no anomalies. Though from a logical point of view a single anomaly is sufficient to demonstrate that there is something wrong with the model that gave rise to it, in practice broader theories are seldom built on curiosities, and anomalies rarely, if ever, play a direct role in the development of better models. It would never occur to a physician to remove a treatment from consideration just because it is known in advance that 3% of the patients will not respond, and it would never occur to a physicist to discard our best model of the universe just because there is a problem with dark matter. This is not to say that the study of low-frequency examples can not be rewarding (just think of the catalytic effect the discovery of Bach-Peters sentences had on semantics) but you have to crawl before you walk – the value of extreme examples becomes clear only against a backdrop of understanding the less extreme cases.

It can hardly be doubted that formal languages (implying divergent weight sums in the Language Model unless describing a finite corpus, a matter we shall turn to in the concluding section) are among the most fruitful abstract models of natural languages ever devised. But to investigate the structural complexity of natural languages, we need not only a mathematical notion of the complexity of the abstract model, as provided e.g. by the Chomsky hierarchy, but also a firm foundation for the idea that the modeling process itself preserves complexity. At present, no such foundation exists for Languages, and linking the complexity of the weights (which themselves can be rational, irrational but algebraic, or even transcendental) to the complexity of the structure is a nontrivial task.

The failure of Chomsky's original argument does not, by itself, invalidate Chomsky's original conclusion that natural languages are outside the simplest (rational) class. However, just as all our numerical models ultimately rest on finite precision arithmetic, it appears that all our effective Language Models can rest on regularly weighted regular languages, and considerations of algebraic complexity, which rarely play a role in numerical work, are destined to remain tangential to the study of natural languages as well.

## 5. Conclusions

It is remarkable that a whole generation of linguists grew up firmly believing in the following Infinity Fallacy (IF):

> No significant generalization about the syntax of a language can rest on a finite class of cases.

Those accepting the IF (and most students of generative syntax still fall in this category) are forced to declare almost every classical branch of linguistics scientifically bankrupt: phonology, morphology, lexicography, and historical linguistics, being primarily concerned with finite corpora, are all suspect. While such revolutionary fervor may have made sense in the early days of generative grammar, by now it seems clear that the bulk of the classical results, e.g. from Indoeuropean, survived the generative revolution essentially intact, while the bulk of early generative grammar had considerably shorter half-life. This suggests that 0-1 models of Languages, and the concomitant use of the finite/infinite distinction as the primary measure for goodness of fit, are simply too crude to deal with the subtler issues that arise in modeling finite data sets.

To the extent that finite data sets remain at the center of the actual practice of linguistics, the more sophisticated quantitative measures discussed in the paper offer a good starting point for linguists interested in developing more numerical methods of argumentation. And to the extent that such argumentation, in particular the creation of models capable of acquiring linguistic generalizations by numerical optimization techniques, is becoming increasingly relevant for applied systems, the mathematical study of languages is also likely to shift from the complexity classes defined in automata-theoretic terms to more information-theoretic considerations of complexity.

## References

J. Beauquier and L. Thimonier: Formal languages and Bernoulli processes. In: J. Demetrovics, G.O.H. Katona, and A. Salomaa (eds):

*Algebra, Combinatorics, and Logic in Computer Science.* North Holland (1986) 97-112

J. Berstel: Sur la densité asymptotique de langages formels. In: M. Nivat (ed): *Automata, Languages, and Programming.* North Holland (1973) 345-358

N. Chomsky: *Syntactic Structures.* Mouton, 1957

N. Chomsky and M.P. Schützenberger: The algebraic theory of context-free languages. In: P. Barfford and D. Hirschberg (eds) *Computer Programming and Formal Systems.* North Holland, 1963, 118-161

S. Eilenberg: *Automata, languages, and machines* vol A, Academic Press, 1974

W. Kuich: On the entropy of context-free languages. *Information and Control* **16** (1970) 173-200

V.H. Yngve: The depth hypothesis. *Language* **61** (1961) 283-305

## Appendix

**Proof of Theorem 1.** Since $r_n \leq k^n$, $d(z) \leq 1/(1-kz)$ so $\rho(L) \leq 1$ will always hold. The transition matrix $A$ associated with the finite deterministic automaton accepting $L$ has column sums $k$, so $B = A/k$ is stochastic. Define $H(z)$ as $(1-z)(E-zB)^{-1}$. The limiting matrix $H = \lim_{z \to 1} H(z)$ always exists, and the density of $L$ is simply $\vec{v}H\vec{e}_i$ where the $j$-th component of $\vec{v}$ is 1 of the $j$-th state is an accepting state (and 0 otherwise) and the initial state is the $i$-th. Since $H(z)$ is a rational function of $z$ and the rational coefficients of $B$, and its values are computed at the rational point $z = 1$, every coefficient of $H$ is rational and so is the density.

**Proof of Theorem 2.** By a suitable rearrangement of the rows and columns of the transition matrix $A$, $B = A/k$ can be decomposed into blocks $D_i$ which appear in the diagonal, a block $C$, which corresponds to transient states and occupies the right lowermost position in the diagonal of blocks, and blocks $S_i$ appearing in the rows of the $D_i$ and the columns of $C$. The column norm of $C$ is less than 1, so $E - C$ can be inverted and its contribution to the limiting matrix is 0. The column sum vectors of $S_i$ can be expressed as linear combinations of the row vectors of $E-C$, and the scalar factors in the theorem are simply the $n$-th coefficients in these expression, where $n$ is the number of the initial state. Moreover, since $(E-C)^{-1} = \sum_{i=1}^{\infty} C^i$ holds, all these scalars will be strictly positive. By the Frobenius-Perron theorem, the density vectors corresponding to the (irreducible) $D_i$ are strictly positive, and this concludes the proof.

**Proof of Theorem 3.** (iii) $\Rightarrow$ (i). If $\rho(L) > 0, \mathcal{A}$ has accepting states in some final class by Theorem 2. If $\alpha \in L$ brings $\mathcal{A}$ in such a state, then no $\beta \in V^*$ can take $\mathcal{A}$ out of this class, and by strong connectedness there is a $\gamma \in V^*$ that takes it back to the accepting state, i.e. $\alpha\beta\gamma \in L$. Thus, $\alpha$ can not be blocked.

(i) $\Rightarrow$ (ii). This is a direct consequence of Theorem 2.

(ii) $\Rightarrow$ (iii). If the accepting states of $\mathcal{A}$ are transient, then for every such state $i$ there exists a string $\beta_i$ that takes the automaton in some state in a final class. Since such classes can not be left and contain no accepting states, the strings $\beta_i$ block the language.

**Proof of Theorem 4.** If $\lim_{z\to 1}(1-z)d(z/k) > 0$, then $d(z/k)$ tends to infinity in $z = 1$, and since it is convergent inside the unit circle, $\sigma$ must be 1. If $L$ is regular, $d(z/k)$ is rational (since it is the result of matrix inversion), therefore if it is bounded in $z = 1$, it has to be convergent on a disk properly containing the unit circle. If $L_1 \subseteq L_2$, then $d_1(z/k) \leq d_2(z/k)$ and since the Taylor coefficients are nonnegative, it is sufficient to look for singularities on the positive half-line. There $d_1(z/k)$ must be convergent if $d_2(z/k)$ is convergent, so $\sigma(L_1) \leq \sigma(L_2)$. Finally, if $d(z/k)$ is convergent on the whole plane, then $f(1) = \sum_{n=0}^{\infty} r_n < \infty$, so $L$ must be finite.


## Acknowledgements

Input from anonymous reviewers is gratefully acknowledged.