Available online at www.sciencedirect.com
SCIENCE DIRECT®

Journal of Combinatorial Theory, Series A 112 (2005) 250–262

Journal of Combinatorial Theory Series A

www.elsevier.com/locate/jcta

Bounds for the growth rate of meander numbers

M.H. Albert[a], M.S. Paterson[b]

[a]Department of Computer Science, University of Otago, Dunedin, New Zealand
[b]Department of Computer Science, University of Warwick, Coventry, UK

Received 22 September 2004

Communicated by David Jackson
Available online 19 April 2005

Abstract

We provide improvements on the best currently known upper and lower bounds for the exponential growth rate of meanders. The method of proof for the upper bounds is to extend the Goulden–Jackson *cluster method*.

MSC: 05A15; 05A16; 57M99; 82B4

Keywords: Meander; Cluster method

1. Introduction

A *meander* of order n is a self-avoiding closed curve crossing a given line in the plane at $2n$ places, [11]. Two meanders are equivalent if one can be transformed into the other by continuous deformations of the plane, which leave the line fixed (as a set). A number of authors have addressed the problem of exact and asymptotic enumeration of the number M_n of meanders of order n (see for instance [6,9] and references therein). The relationship between the enumeration of meanders and Hilbert's 16th problem is discussed in [1] and a general survey of the connection between meanders and related structures and problems in mathematics and physics can be found in [2].

It is widely believed that an asymptotic formula

$$M_n \approx C M^n n^\alpha$$

E-mail addresses: malbert@cs.otago.ac.nz (M. Albert), msp@dcs.warwick.ac.uk (M. Paterson).

0097-3165/$ - see front matter
doi:10.1016/j.jcta.2005.02.006

applies, and some effort has been devoted to estimating the parameters M and α [3–5,10]. Broadly, these methods have relied on extrapolation from exact values of M_n, currently known for $n \leqslant 24$ (see [10]). A careful estimate, using differential approximants based on these values, yields [10] the approximate value $M \simeq 12.26287$.

A presumed correspondence with certain field theories has yielded the conjecture [4] that:

$$\alpha = \sqrt{29}(\sqrt{29} + \sqrt{5})/12 = 3.42013288\ldots .$$

Numerical evidence supporting this conjecture is found in [5]. Our, less ambitious, aim will be to provide rigorous upper and lower bounds on the exponential growth rate of M_n.

Consider the generating function:

$$M(t) = \sum_{n=0}^{\infty} M_n t^{2n}.$$

It is easy to verify that $M_{a+b} \geqslant M_a M_b$ and so it is certainly the case that $M := \lim_{n\to\infty} M_n^{1/n}$ exists, and is the square of the reciprocal of the radius of convergence of this series. We will prove:

Theorem 1.1. *The following inequalities hold*:

$$11.380 \leqslant M \leqslant 12.901.$$

These bounds improve (on both sides) the best previous bounds due to Richard Stanley ($M > 10.0$) [1995, private communication] and Jim Reeds and Larry Shepp ($M \leqslant 13.002$) [1999, unpublished].

Our basic methodology is to represent meanders as a language over an alphabet consisting of four symbols. The bounds are then obtained by producing suitable sublanguages and superlanguages for which the growth rates can be computed explicitly. In principle our bounds could be improved by more detailed construction of these languages, and we include some indication in the final section of how much further progress might be possible by such means.

2. Definitions and notation

We begin by providing a combinatorial description of meanders which allows us to identify them with a language over a four-letter alphabet. This interpretation is implicit in a number of earlier works and is similar to the description of meanders by means of "configurations" in [9] and also to the description of meanders in terms of Motzkin words found in [13].

Set the orientation of the line which the meander crosses as horizontal. We consider the evolution of the meander as we move rightwards along the line. When viewed in this way, at a particular stage of its evolution the meander will consist of a number of segments, some of whose endpoints lie above the line, and some which lie below. Each significant step in the

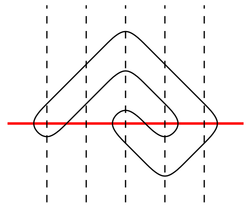

Fig. 1. The meander cucdaa.

evolution of a meander is marked by a place where the meander crosses the line, and these crossings are of four types: c where a new segment of the meander is created, a where two previous segments are merged into one (or as a final step the meander is completed), and u or d where a segment crosses the line from bottom to top, or top to bottom, respectively. Fig. 1 illustrates this encoding of meanders.

The *meander language*, $\mathcal{M}$, is the set of words in these four letters that represent meanders. It is immediately clear that distinct words in the meander language represent distinct meanders, and only slightly less clear that every meander is represented by a single word in the meander language.

We digress briefly to recapitulate some standard notation and terminology concerning words and languages. A *word* is simply a finite sequence of symbols from some alphabet Σ. This sequence may be empty, and the empty word is denoted ε. The set of all words over Σ is denoted Σ^* and can be identified with the free monoid over Σ by considering juxtaposition as the monoid operation. So, a word v is said to be a *factor* of a word w if $w = xvy$ for some words x and y. If we can take $x = \varepsilon$ then we say that v is a *prefix* of w, while if we can take $y = \varepsilon$ then we say that v is a *suffix* of w. A *language* over Σ is simply a subset of Σ^*. The $()^*$ notation is extended to languages, or even words, so that X^* simply means the language which consists of all possible juxtapositions (including the empty one) of elements of X. The length of a word w, that is, the number of symbols in the sequence w, is denoted $|w|$. Hence M_n, the number of meanders with $2n$ crossings is simply the number of words in $\mathcal{M}$ of length $2n$ (since each symbol in a meander word accounts for a single crossing).

In our interpretation of meanders it makes sense to speak of the environment that exists as we scan prefixes of a word. This environment is simply the collection of segments in their appropriate order on either side of the line. Further, we adopt the convention that when two segments are merged, the newly merged segment is identified in the environment with the older of the two (in a meander the only time we will merge two ends of the same segment is at the final a).

Sometimes it is useful to imagine that we have available an extended environment consisting initially of an infinite family of labelled and completely unmatched segments on either side of the line. This allows the effect of any word to be interpreted within this environment. For our purposes, words whose only effect is to shift some segments from one side of the line to the other are particularly significant. In Fig. 2 we illustrate how the factor cuad has no effect on the surrounding environment. In particular this means that if $w = uv$ is a meander, and if the environment following u contains a segment below the line, then $u$cuad$v$ is also

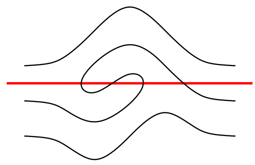

Fig. 2. cuad has no effect on the environment.

a meander. On the other hand, it is also clear that no meander (aside from ca) can have ca as a factor, and so neither can it have ccuada as a factor. From observations of the former kind we obtain sublanguages of $\mathcal{M}$ by building up words which must be meanders. From observations of the latter kind we obtain superlanguages of $\mathcal{M}$ by requiring words to avoid certain factors.

Throughout the remaining sections we identify languages over c, a, u, d with their generating function in the power series ring over non-commuting variables c, a, u and d. Generally we work in this context to obtain relationships between (the generating functions of) various languages, and then specialize to a single variable t when we wish to obtain numerical estimates. All the generating functions we will be considering have non-negative coefficients, and when we say that $f(t)$ *majorizes* $g(t)$ we mean that the coefficient of t^n in $f(t)$ is at least as great as that in $g(t)$. In particular, this implies that the radius of convergence of f is not greater than that of g.

3. Shifts and lower bounds

Consider a state of the extended meander environment, such as might be achieved after executing some prefix p of a meander word. There are now various continuations which will have the same effect *on the environment* as u^k would for some k. Trivially any sequence of u's and d's which has k more u's than d's is such a continuation. However, it is also the case that cua has the same effect on the environment as u, and ccuuaa has the same effect as uu. Furthermore, these constructions can be recursively combined and therefore:

c(ccuuaa)da

has the same effect as cuuda, hence as cua and finally as u.

Definition 3.1. A *shift* is a word whose effect on the extended meander environment is the same as that of u^k or d^k for some non-negative integer k. The *displacement* of a shift is k in the former case, and $-k$ in the latter. A *jump* is a shift having no proper shift prefix.[1] A shift whose only proper shift factors are in u^* or d^* is called *primitive*.

The simplest jumps are u and d. Next simplest are cua and cda. A rather more complicated example is shown in Fig. 3.

[1] We apologize to the sensitive reader for using “shift” both as a noun and an adjective.

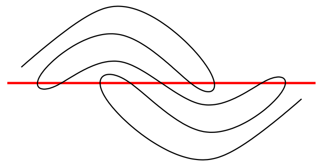

Fig. 3. $\mathsf{cucd}^3\mathsf{aua}$ is a jump of displacement -1.

Every shift can be uniquely factored as a concatenation of jumps. In turn, every jump is created from some (uniquely determined) primitive shift by substitution of shifts for the blocks of u's and d's within the primitive shift. For example cccuuaada is created from cua by substituting ccuuaa d (a shift of displacement 1 formed from a jump of displacement 2, and one of displacement -1) for u.

Every shift has a *minimum required environment*. This is simply the minimum number of segments which must be present on either side of the line before undertaking the shift in order to ensure that it can be successfully carried out. As previously noted, the minimum required environment for the shift cua is that there be at least one segment below the line (which will be merged with the end of the segment created at the beginning of the jump). On the other hand the shift $\mathsf{u}^{100}\mathsf{d}^{200}\mathsf{u}^{100}$ has a minimum required environment of 100 segments above and 100 segments below the line. Any extra segments beyond those of the minimum required environment do not hinder the interpretation of a shift.

Let $\mathcal{J}$ be the language of all jumps and $\mathcal{S}$ the language of all shifts. Since every shift is uniquely represented as a concatenation of jumps, then $\mathcal{S}$ is the union of the disjoint sets $\mathcal{J}^n$ as n runs over the non-negative integers, with the usual convention that $\mathcal{J}^0$ denotes the singleton set consisting of the empty word. Since concatenation of words corresponds to multiplication of the corresponding generating functions, when we pass to generating functions we obtain

$$\mathcal{S} = \mathcal{J}^* = \varepsilon + \mathcal{J} + \mathcal{J}^2 + \cdots = \frac{1}{1-\mathcal{J}}. \tag{1}$$

Introducing a new indexing variable x which commutes with the symbols of the language, and letting $\mathcal{J}_i$ (or $\mathcal{S}_i$) be the language of jumps (or shifts) of displacement i, we have slightly more generally that:

$$\sum_{i=-\infty}^{\infty} \mathcal{S}_i x^i = \frac{1}{1-\sum_{i=-\infty}^{\infty} \mathcal{J}_i x^i}.$$

Suppose that J is some primitive jump. Then the set of all jumps with primitive form J is obtained by replacing each (possibly null) block of d's or u's between consecutive occurrences of c or a by $\mathcal{S}_k$ where k is the displacement of the block. Denote the result of this replacement by $J^{\mathcal{S}}$. Then $\mathcal{J}_i$ is the sum over primitive jumps J of displacement i of the terms $J^{\mathcal{S}}$.

Let $j_0(t)$ be the generating function obtained from $\mathcal{J}_0$ by replacing all of c, a, d, and u by t.

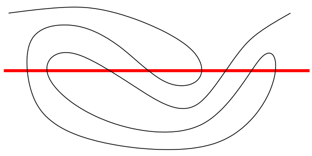

Fig. 4. ccddauua is in $\mathcal{J}_0$.

Proposition 3.2. *The radius of convergence of $j_0(t)$ is equal to that of $M(t)$.*

Proof. First suppose that $r > 0$ is larger than the radius of convergence of $j_0(t)$. Then we may choose a polynomial $j(t)$ with non-negative integer coefficients majorized by those of $j_0(t)$ and such that $j(r) > 1$. We may further choose a finite subset, $\mathcal{F} \subseteq \mathcal{J}_0$ such that the generating function of this finite set is j. Finally, let N be a positive integer larger than the largest number of segments in the minimum required environment of any jump in $\mathcal{F}$. Note that $w = \mathsf{c}^N\mathsf{da}^{N-1}\mathsf{ua} \in \mathcal{M}$ as it represents a large closed spiral with the line running through it. The environment present after the prefix c^N of w consists of N segments on either side of the line. In this environment, any of the jumps in $\mathcal{F}$ can be carried out. In fact, since these jumps all have displacement 0, after carrying one out the environment is restored to its previous state and therefore any word from $\mathcal{F}^*$ can be carried out in this environment. Thus,

$$\mathsf{c}^N\mathcal{F}^*\mathsf{da}^{N-1}\mathsf{ua} \subseteq \mathcal{M}. \tag{2}$$

Moreover, since every word in $\mathcal{F}^*$ factors uniquely as a sequence of words in $\mathcal{F}$ we may pass from Eq. (2) directly to the corresponding generating functions and see that $M(t)$ majorizes

$$\frac{t^{2N+2}}{1 - j(t)}.$$

Therefore, the radius of convergence of $M(t)$ is not greater than the first positive root of $1 - j(t)$, so the radius of convergence of $M(t)$ is smaller than r, and thus less than or equal to the radius of convergence of $j_0(t)$.

Conversely, if $w = c\mathsf{a}$ is any word in $\mathcal{M}$ then

$$\mathsf{c}c\mathsf{ddauua} \in \mathcal{J}_0.$$

Indeed, in order to establish this fact, it suffices to verify it for the case $c = \mathsf{c}$ since that is the net effect of c, and this is illustrated in Fig. 4. Hence, $j_0(t)$ majorizes $t^6M(t)$. Thus, the radius of convergence of $j_0(t)$ does not exceed that of $M(t)$ and so, in conjunction with the result of the previous paragraph, we may conclude that they must be equal. □

This proposition is our main tool in computing lower bounds for the constant M. It allows us to take any polynomial majorized by $j_0(t)$ and use it to produce an upper bound on the radius of convergence of $M(t)$ and hence a lower bound on M. This method can be used effectively because we can generate many jumps from primitive jumps by successive substitutions of shifts within them, and also evaluate the corresponding generating functions at a particular value $r > 0$ by corresponding substitutions of previously computed values of the generating functions. The details of this method are discussed further in Section 6.

4. The cluster method

The *cluster method* is a method of enumerating words with a given finite set of forbidden factors. It was introduced in this form in [7] and is also discussed in [8]. Extensions of the cluster method are given in [12] to handle certain cases where the forbidden set of factors is infinite. We need to supply a similar extension in an even more general setting.

Let Σ be an alphabet, and $\mathcal{B}$ a subset of Σ^+ (the non-empty words over Σ). We are concerned with the language consisting of those words which have no factor from $\mathcal{B}$, the *$\mathcal{B}$-factor-free words*, that is the complement in Σ^* of $\Sigma^*\mathcal{B}\Sigma^*$. If b is a factor of c and b does not occur as a factor of some word w, then of course neither does c. So, for any $\mathcal{B}$, the $\mathcal{B}$-factor-free words are the same as the $\mathcal{B}'$-factor-free words, where $\mathcal{B}'$ consists of the minimal elements of $\mathcal{B}$ in the factor ordering. Therefore we assume throughout that no word $b \in \mathcal{B}$ is a proper factor of any other word in $\mathcal{B}$.

Define the set of *overlaps*, $Ov(\mathcal{B})$ to be the collection of all triples (b, w, c) such that $b, c \in \mathcal{B}$, $w \in \Sigma^+$, and for some b^l and c^r, $b = b^l w$ and $c = wc^r$. Note that, owing to the assumption above, neither b^l nor c^r can be the empty word. We claim that the system of equations:

$$v_b = b - \sum \left\{ b^l v_c \;:\; (b, w, c) \in Ov(\mathcal{B}),\; b = b^l w \right\} \qquad \text{for } b \in \mathcal{B} \tag{3}$$

has a unique solution in the power series ring $\mathbb{Q}[[\Sigma]]$.

For consider the linear operator:

$$S \;:\; \mathbb{Q}[[\Sigma]]^{\mathcal{B}} \;\to\; \mathbb{Q}[[\Sigma]]^{\mathcal{B}}$$

that sends x to Sx where:

$$(Sx)_b = \sum \left\{ b^l x_c \;:\; (b, w, c) \in Ov(\mathcal{B}),\; b = b^l w \right\}.$$

Then, if the minimum degree of any monomial occurring in any coordinate of x is d, the minimum such degree in Sx is strictly larger than d. Therefore S has norm less than 1 with respect to a suitable valuation of $\mathbb{Q}[[\Sigma]]$, and the L^∞ norm on $\mathbb{Q}[[\Sigma]]^{\mathcal{B}}$. It follows that $I + S$ is invertible, and hence that the system of equations (3) has a unique solution.

The following theorem generalizes (to the case of infinite $\mathcal{B}$ and non-commuting variables) a specialization (to the case of forbidding all occurrences of $\mathcal{B}$ rather than determining the type of the occurrences of $\mathcal{B}$ in a word) of Theorem 2.86 in [8], often called the Goulden–Jackson cluster method. In [14] an informal treatment of an equivalent method can also be found. A full generalization of the original theorem could be obtained by adding tagging variables y_b (commuting with each other and with Σ) to the system (3), but the version below is adequate for our purposes.

Theorem 4.1. *The generating function over* $\mathbb{Q}[[\Sigma]]$ *of* $\Sigma^* \setminus \Sigma^* \mathcal{B} \Sigma^*$ *is*

$$\left(1 - \Sigma + \sum_{b \in \mathcal{B}} v_b\right)^{-1},$$

where $\{v_b \,:\, b \in \mathcal{B}\}$ *are defined by* (3).

Proof. The proof of this theorem can be read off from the proof of the theorem cited above. However, at least in this form, it is simply a restatement of the principle of inclusion/exclusion. Define a $\mathcal{B}$-marking of a word w in Σ^* to be a specific identification of certain factors of w which belong to $\mathcal{B}$ (not necessarily any or all such factors). If we assign the value $(-1)^k w$ to each $\mathcal{B}$-marking of w in which k factors from $\mathcal{B}$ are marked then the sum over all the $\mathcal{B}$ markings of a word w will be 0 if w contains a $\mathcal{B}$-factor, and w if it does not. By considering the expression above as a geometric series it is easy to see that the coefficient of w is exactly this sum over $\mathcal{B}$-markings of w, and hence the expression represents the generating function of $\mathcal{B}$-factor-free words. $\square$

As remarked in [14], in the case of infinite structureless $\mathcal{B}$ this does not give an equation for the generating function in any usual sense. However, in our application below, the language $\mathcal{B}$ will carry sufficient structure that we can make effective use of Theorem 4.1.

Note that if we turn to the ordinary generating function for the language of $\mathcal{B}$-factor-free words, then its radius of convergence is the smallest positive root of the equation:

$$1 - |\Sigma| t + \sum_{b \in \mathcal{B}} v_b(t),$$

where we also have

$$v_b(t) = t^{|b|} - \sum \left\{ t^{|b^l|} v_c(t) \,:\, (b, w, c) \in Ov(\mathcal{B}),\ b = b^l w \right\} \qquad \text{for } b \in \mathcal{B}.$$

Remark 4.2. The system of equations (3) can be solved directly by an iterative method beginning from $v_b = 0$ for all $b \in \mathcal{B}$. In general, however, the system of linear equations defined above fails to have the property required to allow an iterative solution after specializing to a single variable, and to a specific value for that variable, even if the value chosen lies inside the radius of convergence of the series which form its solution in $\mathbb{Q}[[t]]$. An example is the case $\mathcal{B} = \{aaa, aba\}$ over the alphabet $\{a, b\}$.

5. Submeanders and upper bounds

We now apply the results of the preceding section in order to obtain upper bounds on the exponential growth rate of the meander language $\mathcal{M}$. Ideally, the language of forbidden words which we would like to consider consists of all words which minimally define some closed loop, or *submeander*. That is, a word is forbidden if it is of the form $\mathsf{c}w\mathsf{a}$ where the final symbol closes off the pair of segments created by the initial one, and such that w does not create a submeander. Let $\mathcal{B}$ be the language of such words. If an element of $\mathcal{B}$ occurs as a *proper* factor of a word m then $m \notin \mathcal{M}$. However, by definition, $\mathcal{M} \subseteq \mathcal{B}$ as well, so the words in $\mathcal{M}$ are not $\mathcal{B}$-factor-free. But the growth rates for the languages of $\mathcal{B}$-factor-free words and proper $\mathcal{B}$-factor-free words are the same, so we do not need to worry about that distinction. Henceforth we fix the alphabet $\Sigma = \{\mathsf{c}, \mathsf{a}, \mathsf{u}, \mathsf{d}\}$.

The shortest word in $\mathcal{B}$ is ca. However, this single word is really a representative of a much wider family of forbidden words. Among these are cuda, and ccuada. Generally if S is any shift of displacement 0, then cSa is a forbidden word. It is worth noting that there is no requirement that the words in $\mathcal{B}$ be balanced with respect to c and a. For example, the word cucdda is in $\mathcal{B}$, since the final a forms a submeander with the original c, and so if this word occurs as a factor of some longer word w then w cannot represent a meander.

There is an equivalence relation defined on words by taking the transitive closure of the relation obtained by allowing the replacement of a shift by any other shift of the same displacement. Each equivalence class of this relation contains a unique representative with the property that any maximal shift factor lies in d^* or u^*. Let us call these representatives the *standard representatives* of their classes. Note that $\mathcal{B}$ is closed under this equivalence relation.

Lemma 5.1. *Let a word w be given. Its standard representative is obtained by replacing the maximal shift factors of w by blocks of* d*'s or* u*'s of the same displacement.*

Proof. This follows immediately from the observation that two shift factors of w cannot overlap unless their overlap is also a shift. This is because a proper suffix of a shift which is not a shift and begins with c contains more a's than c's, and no prefix of a shift word has this property. Since shifts are closed under concatenation, the maximal shift factors of w are disjoint and properly separated, and so the standard representative is obtained in the manner described. □

Using this result we obtain:

Proposition 5.2. *Let $b, c \in \mathcal{B}$ have an overlap w. Then the standard representatives of b and c also have an overlap, which is the image of w under the replacement described in Lemma* 5.1.

Proof. The word w has the form cua. Moreover in b the terminal a closes the segments formed by the initial c of b so, interpreted in isolation, it does not close any segment created within u and so cannot be part of any shift factor of w. The same idea applies to the observation that the initial c of c is matched by its final a and so shows that the original c of w also cannot be part of any shift factor of w. So the shift factors of b and c which occur within w, occur within u. Therefore the reduction of Lemma 5.1 affects w in the same way in both b and c. □

Let $\mathcal{B}_{\mathsf{rep}}$ be the sublanguage of $\mathcal{B}$ consisting of the standard representatives of the elements of $\mathcal{B}$. For any word w let $\bar{w}$ be the generating function of its equivalence class. Now consider a modification of the system of equations (3):

$$x_b = \bar{b} - \sum \left\{ \bar{b^l} x_c \ :\ (b, w, c) \in Ov(\mathcal{B}_{\mathsf{rep}}),\ b = b^l w \right\} \qquad \text{for } b \in \mathcal{B}_{\mathsf{rep}}. \tag{4}$$

Then, it follows directly from Proposition 5.2 that:

$$\sum_{b\in\mathcal{B}} v_b = \sum_{b\in\mathcal{B}_{\mathsf{rep}}} x_b,$$

where v_b is defined by the system of equations (3).

Thus we may use the modified form in computations arising from Theorem 4.1. For instance, we could use a finite subset of the original language $\mathcal{B}$, and also place some restrictions on the shift words used in constructing $\bar{w}$ from w.

For example, take as forbidden language $\mathcal{B}_0$, the single forbidden word ca and its expansions $\mathsf{c}S\mathsf{a}$ where $S \in \{\mathsf{u},\mathsf{d}\}^*$ has displacement 0. Then the generating function for $\mathcal{B}_0$-factor-free words is

$$\frac{1}{1-4t+\dfrac{t^2}{\sqrt{1-4t^2}}}.$$

The radius of convergence of this generating function is the smallest positive solution of

$$65t^4 - 32t^3 - 12t^2 + 8t - 1 = 0,$$

whose approximate value is 0.272054. Since $\mathcal{B}_0$ represents a subset of the actual words forbidden to appear as factors in a meander word, this gives an upper bound of 13.5111 on M.

In the next section we will describe in greater detail how these results can be used to provide bounds for M in situations where we cannot analytically solve the equations for the radius of convergence.

6. Computational methodology

In this section we give an overview of the computational methods used to evaluate lower and upper bounds on M.

6.1. Lower bounds

In computing lower bounds on the exponential growth rate for the meander generating function, we attempt to construct a generating function based on a subset of the set of shifts, built up from a subset of the primitive jumps. Generally, we make use of all the primitive jumps containing at most some preset number of symbols. These are constructed by simulating the extended meander environment and carrying out a depth-first search. The only extra information which must be maintained is a record of the new segments present when each c occurs. This must then be compared to the a which eliminates the segment created by the c in order to ensure that the only shift factors are in d^* and u^*.

The results quoted below are for primitive jumps containing a maximum of 24 symbols. There are 875,938 such primitive jumps with non-negative displacement. On the other hand, there are only 25,264 of length at most 20, and only the following 13 of length at most 10:

cua	ccuuaa	ccddauua	ccuuadda
cddcuuaa	cuucddaa	cccuuuaaa	cdcuuuada
cdccuuauaa	cuucddauua	ccucuuaada	ccdcuuuaaa
cccuuuadaa.			

The basic computational scheme employed is a simple iterative one. We establish at the outset an arbitrary bound on the number of jumps which will be concatenated to form a shift (in practice 50 is more than adequate). Then we take an existing set of jumps and compute a new set of shifts by concatenating them in this way. These new shifts are in turn substituted into our supply of primitive jumps in order to compute a new set of jumps, and so on.

All of this is handled numerically by passing at the outset to generating functions in a single variable t (which replaces each of the letters of the meander alphabet). For a fixed real value of t we can then carry out the computation described above. At the end of each iteration we have a value $j(t)$ that represents the value of the generating polynomial for the jumps of displacement 0 constructed up to this point. Using Proposition 3.2 we can conclude that if $j(t) > 1$ then t lies outside the radius of convergence of the generating function of the meander language. So this decision is a strict one—we can be certain that the upper bounds supplied for the radius of convergence are accurate. For the lower bounds we simply continue the iteration for some preset number of steps, and declare that we are inside the radius of convergence if we have never found $j(t) > 1$. Then a simple binary search on t allows us to determine rigorous upper bounds on the radius of convergence for the meander language.

Using jumps of length up to 24, we obtain an upper bound of 0.296431 for the radius of convergence of the generating function of the meander language. This translates to a lower bound of 11.38 on M.

6.2. Upper bounds

In producing upper bounds for the growth rate of meander numbers we begin from a set $\mathcal{B}$ of standard representatives of words creating a submeander. Again, the most straightforward approach is simply to list all such words up to some predefined length. Doing this, again involves a depth-first search in the extended meander environment. This time we must check that the final a joins the segments formed by the initial c, that no earlier a creates a submeander, and that no jumps occur as subwords other than d and u. All these tests are easily implemented within the meander environment.

After passing to a single variable t we use Eq. (4) in order to compute the quantities x_b. Rather than solving this large (but relatively sparse) system exactly we may use a simple iterative scheme, since it is easily checked that for values of t in the range we are interested in, there are no eigenvalues of the matrix representing the summations on the RHS of this equation whose modulus is greater than or equal to 1. Convergence is therefore guaranteed, with error bounds decreasing by a constant factor on each iteration. Having computed the

Table 1
Lower and upper bounds on M based on maximum length of jumps, and submeanders

Lower bounds		Upper bounds	
10	10.749	6	13.171
12	10.928	8	13.086
14	11.023	10	13.018
16	11.114	12	12.970
18	11.188	14	12.931
20	11.249	16	12.901
22	11.301		
24	11.380		

values x_b, all that is necessary is to evaluate the sign of

$$1 - 4t + \sum_{b \in \mathcal{B}} x_b(t),$$

which is the denominator of the generating function for the $\mathcal{B}$-factor-free words found in Theorem 4.1. This allows us to determine whether t lies above or below the radius of convergence of the generating function for the $\mathcal{B}$-factor-free words. Again a simple binary search can be used to estimate the radius of convergence, and hence an upper bound on the exponential growth of the meander numbers.

Using the 20 509 words of length 16 which are standard representatives of words creating a submeander for $\mathcal{B}$ produces an estimate of 0.2784 for the radius of convergence of $\mathcal{B}$-factor-free words, and hence an upper bound of 12.901 on M.

7. Summary and conclusions

Obviously the methods which we have applied could be extended to obtain better bounds through more extensive computation using longer words as primitive jumps, or as the standard representatives of submeander words. Some indication of how far this might or might not progress is shown in Table 1.

A simple extrapolation based on this data suggests a limiting lower bound of approximately 11.6, and an upper bound of approximately 12.8. However, the final lower bound which we have computed (from jumps up to length 24) represents a better than expected improvement on the previous value. Put another way, there are more jumps of length 24 than one would expect based on simple extrapolation of previous values. So, it may be that better improvements on the lower bound are possible.

Acknowledgments

This material was presented at FPSAC 2004. We would like to thank the referees of our conference submission who made many valuable suggestions that have improved the presentation of this paper.

References

[1] V.I. Arnol'd, The branched covering $\mathbf{C}P2 \to S4$, hyperbolicity and projective topology, Sibirsk. Mat. Zh. 29 (5) (1988) 36–47, 237.
[2] P. Di Francesco, Folding and coloring problems in mathematics and physics, Bull. Amer. Math. Soc. (N.S.) 37 (3) (2000) 251–307 (electronic).
[3] P. Di Francesco, Exact asymptotics of meander numbers, in: Formal Power Series and Algebraic Combinatorics, Moscow, 2000, Springer, Berlin, 2000, pp. 3–14.
[4] P. Di Francesco, O. Golinelli, E. Guitter, Meanders: exact asymptotics, Nucl. Phys. B 570 (3) (2000) 699–712.
[5] P. Di Francesco, E. Guitter, J.L. Jacobsen, Exact meander asymptotics: a numerical check, Nucl. Phys. B 580 (3) (2000) 757–795.
[6] R.O.W. Franz, B.A. Earnshaw, A constructive enumeration of meanders, Ann. Comb. 6 (1) (2002) 7–17.
[7] I.P. Goulden, D.M. Jackson, An inversion theorem for cluster decompositions of sequences with distinguished subsequences, J. London Math. Soc. (2) 20 (3) (1979) 567–576.
[8] I.P. Goulden, D.M. Jackson, Combinatorial Enumeration, A Wiley-Interscience Publication, Wiley, New York, 1983 with a foreword by Gian-Carlo Rota, Wiley-Interscience Series in Discrete Mathematics.
[9] I. Jensen, A transfer matrix approach to the enumeration of plane meanders, J. Phys. A: Math. Gen. 33 (34) (2000) 5953–5963.
[10] I. Jensen, A.J. Guttmann, Critical exponents of plane meanders, J. Phys. A: Math. Gen. 33 (21) (2000) L187–L192.
[11] S.K. Lando, A.K. Zvonkin, Plane and projective meanders, Theoret. Comput. Sci. 117 (1–2) (1993) 227–241 Conference on Formal Power Series and Algebraic Combinatorics, Bordeaux, 1991.
[12] J. Noonan, D. Zeilberger, The Goulden–Jackson cluster method: extensions, applications and implementations, J. Differential Equations Appl. 5 (4–5) (1999) 355–377.
[13] A. Panayotopoulos, P. Tsikouras, Meanders and Motzkin words, J. Integer Seq. 7 (1) (2004) 10 Article 04.1.2 (electronic).
[14] D. Zeilberger, The umbral transfer-matrix method. V. The Goulden–Jackson cluster method for infinitely many mistakes, Integers 2 (2002) 12, Paper A5 (electronic).