Self-avoiding rings on the triangular lattice

1992 J. Phys. A: Math. Gen. 25 2791

(http://iopscience.iop.org/0305-4470/25/10/009)

J. Phys. A: Math. Gen. 25 (1992) 2791–2807. Printed in the UK

Self-avoiding rings on the triangular lattice

I G Enting† and A J Guttmann
Department of Mathematics, The University of Melbourne, Parkville, Vic 3052, Australia

Received 12 December 1991

Abstract. The finite lattice method of series expansion is used to extend the enumeration of self-avoiding polygons on the triangular lattice through to rings of 35 steps. We also give the enumeration of these triangular polygons grouped by perimeter and area of up to 21 unit triangles for perimeters up to 24 steps.

1. Introduction

The enumeration of self-avoiding rings (polygons) on regular lattices has proved to be one of the more difficult problems in lattice statistics. The only exactly known results are the connective constant and critical exponents for polygons on the honeycomb lattice (Nienhuis 1982, 1984). The exponents $\alpha = \frac{1}{2}$ and $\nu = \frac{3}{4}$ are assumed to apply to all other two-dimensional lattices. A number of inequalities involving polygon statistics are known but essentially all other information on the problem comes from the analysis of enumerations of finite polygons, i.e. from series expansions of the polygon generating function.

In two dimensions, a major improvement in the derivation of polygon series was made by Enting (1980) who extended the square lattice enumeration of polygons from 24 steps to 38 steps using a finite lattice method. This enumeration was extended to 46 steps by Enting and Guttmann (1985) who also enumerated polygons of up to 48 steps on the L and Manhattan lattices. Using the same technique, Guttmann and Enting (1988) extended the square lattice series to 56 steps. This work also gives series for the spanning moments of polygons on the square lattice. More recently, Enting and Guttmann (1989) have extended the polygon series on the honeycomb lattice from 34 steps to 82 steps, and most recently we have enumerated square lattice polygons by both perimeter and area for perimeters up to 42 and areas up to 20 (Enting and Guttmann 1990).

In the various applications of the finite lattice method, the honeycomb, L and Manhattan lattices were treated as sublattices of the square lattice and so the original finite lattice algorithm of Enting (1980) could be applied with relatively minor modifications. However it has not been possible to extend the triangular lattice polygon series by treating the lattice as a square lattice with some diagonal bonds with the enumeration algorithm otherwise unchanged. Such an approach can sometimes be useful for low-temperature series (see for example Enting and Wu 1982) but is

† Permanent address: CSIRO, Division of Atmospheric Research, Private Bag 1, Mordialloc, Vic 3195, Australia.

0305-4470/92/102791+17$04.50

less suitable for high-temperature series. For high-temperature series (which polygon series closely resemble), the inclusion of extra bonds increases the computational requirements to such an extent that the original form of the finite lattice method of polygon enumeration could barely compete with direct graph counting.

The use of the finite lattice method to enumerate polygons involves two distinct combinatorial problems. The first problem is the enumeration of classes of polygon embedded in the finite lattices. The second problem is that of combining the enumerations from finite lattices to produce the (truncated) generating function for the infinite lattice limit. In using the finite lattice method to enumerate polygons on the triangular lattice, both these combinatorial problems must be approached differently from the square lattice case. Firstly, the enumeration of polygons on the finite lattice uses a 'site' representation rather than the 'bond' representation used in all previous work. This new approach is described in section 2; this change enabled us to extend earlier series (Sykes *et al* 1972) from 18 to 25 steps as reported by Enting and Guttmann (1990). Secondly, in the present work, the finite lattice generating functions are evaluated for various hexagonal finite lattices rather than the rectangles used in our previous calculations. The combinatorial relations for combining such finite lattice generating functions are described in section 3. These relations are generalizations of those described by Enting (1987), modifed to take account of the special way in which the finite lattice generating functions for polygons are defined. Combining these two extensions of the finite lattice method has enabled us to extend the perimeter generating function for polygons on the triangular lattice to 35 steps. Section 4 presents an analysis of the resultant series.

2. Finite lattice enumerations of polygons

The finite lattice method requires the solution of lattice statistics problems on finite lattices. For lattice statistics problems such as the Potts model, the local nature of the interaction makes it easy to build up a finite lattice one site at a time. The polygon enumeration problem is one of greater difficulty. If one considers the structure of the polygon in an isolated region (a Eulerian representation in kinematic terms) then the requirement that a set of bonds form a closed ring is a non-local constraint. If one follows the steps of the polygon (a Lagrangian representation) then the self-avoidance is a non-local constraint. However the polygon enumeration problem does exhibit some degree of localization in the constraints. This is shown most easily in two dimensions. If one draws a transect line cutting the polygon as shown in figure 1 then

(i) the self-avoidance constraint acts independently on each side of the transect line; and

(ii) if the graph is to be a self-avoiding polygon then the connectivity of the loops on one side of the transect line uniquely specifies the connectivity of the loops on the other side of the transect line.

These two properties are sufficient to enable us to enumerate polygons using a transfer matrix method. On constructs a generating function $\mu_\nu(T)$ which enumerates all polygon segments to the left of line, T, that intersect T in a specific manner identified by the index ν. If one has two transect lines S and T, as shown in figure 1, and $M_{\nu'\nu}(S,T)$ is the number of ways in which a segment intersecting T with

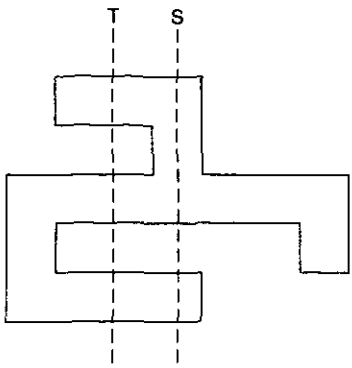

Figure 1. Transect lines (broken) intersecting an arbitrary polygon.

index ν can be extended to intersect S with index ν' then

$$\mu'_{\nu}(S) = \sum_{\nu} M_{\nu'\nu}(T,S)\,\mu_{\nu}(T). \tag{2.1a}$$

Moving a transect line in a series of steps $T_1, \ldots, T_N$ gives

$$\mu_{\nu_N}(T_N) = \sum_{\nu_{n-1}} \cdots \sum_{\nu_1} M_{\nu_N\nu_{N-1}}(T_N, T_{N-1}) \cdots M_{\nu_2\nu_1}(T_2, T_1)\,\mu_{\nu_1}(T_1) \tag{2.1b}$$

so that the generating function for polygons in a finite region can be constructed from a null starting value by a series of operations multiplying a state vector μ_{ν} by a succession of matrices. If the successive transect lines are almost identical so that only a single site is added at each step, then the matrices $M_{\nu\nu'}(S,T)$ become very simple and extremely sparse and so need not be stored explicitly. The computational effort then grows in proportion to the size of the state vectors. When classifying polygons in terms of their numbers, m, of steps, a variable x^m is included in the definitions of the state vectors and transfer matrices so that the components $\mu_{\nu}(T)$ are truncated series in x. Similarly a variable y^n can be included to classify polygons in terms of area.

In previous applications of the finite lattice method, the transect lines were drawn on the dual lattice so as to pass through the centre of each bond. Inspection of figure 2 shows that for a given width of lattice, applying such an approach on the triangular lattice would result in about twice as many bond intersections as on the square lattice. Therefore, the dimension of the state vector for the triangular enumeration would be approximately the square of the corresponding square lattice state vector dimension. To avoid such a catastrophic increase in complexity we use transect lines that effectively split sites as shown in figure 3. This leads us to consider four types of points of (possible) intersection between polygon segments and the transect line as shown schematically in figure 4. Type 0 is a point on the transect line at which the left-hand segment of the polygon is absent. Types 1 and 2 are points at which the polygon crosses the transect line and type 3 is a point at which the left-hand segment touches the transect line but does not cross it. As in the square lattice case, types 1 and 2 are distinguished so that the connectivity of the segment is specified. Each loop will cross the transect line twice. The upper intersection is of type 1 and the lower intersection is of type 2. Thus, as shown by Enting (1980), the use of labels '1'

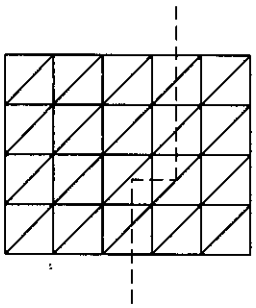

Figure 2. If the transect line is drawn on the dual lattice, then for fixed width w, the triangular lattice has almost twice as many intersections as the square lattice.

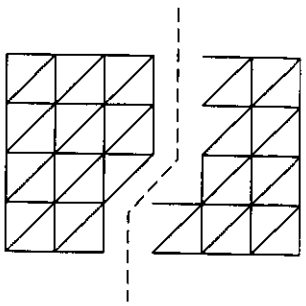

Figure 3. Schematic representation of how the transect line used in this work cuts through sites rather than bonds.

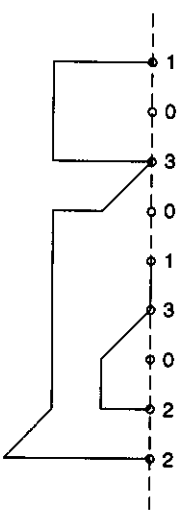

Figure 4. *Examples of configurations on intersections* of the left side of the polygon and the transect line.

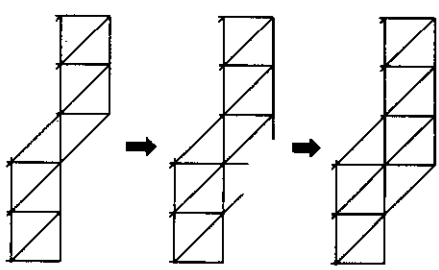

Figure 5. The two stages of adding a site to the finite lattice. Firstly three new bond positions are occupied in each allowed way and assigned types '1' or '2' according to table 2. States of sites 'a' and 'c' are changed as specified in table 2. Secondly the new site 'b' is assigned a type, according to table 4 and the resulting contribution to the partial generating function is processed as indicated in table 4.

and '2' on bonds that cross the transect line is sufficient to specify the connectivity of the polygon segment to the left of the transect line.

In building up generating functions for polygons, we need to consider all possible combinations of intersection labels and map these onto an index set $\{\nu\}$. The labels n_j are assigned to sites 0 to W and take values 0, 1, 2, 3. If we define

$$\sigma_k(M) = \sum_{j \leqslant k} \delta_{M,n_j} \tag{2.2a}$$

then we require

$$\sigma_j(1) \geqslant \sigma_j(2) \qquad \text{for all } j \tag{2.2b}$$

and

$$\sigma_W(1) = \sigma_W(2). \tag{2.2c}$$

The constraints on the '1' and '2' labels are the same as in our earlier work. The difference is that in the present case we allow the '3' state. Table 1 lists the number of allowed combinations of labels for various values of W and also shows the order to which the triangular lattice generating function can be evaluated if strips of width $\leqslant W$ are used.

Table 1. The number of elements, $|\mu(T)|$, in the state vector, $\mu(T)$ and the order to which triangular lattice polygons can be obtained using square lattice and triangular lattice inclusion/exclusion relations, as a function of strip width W. $2W+3$ gives the maximum perimeter of polygons that can be enumerated using strips of width W in the rectangular symmetry formalism and incorporating the correction of 2 in the last term. $p = 3W+2$ gives the maximum perimeter of polygons that can be enumerated using lattices of width W and the triangular symmetry formalism. $|A(p)|$ gives the cumulated number of hexagons with perimeter $\leqslant p$. $|\{a(n)\}|$ is the cumulated number of canonical hexagons and $|\{b(n)\}|$ is the number of hexagons with non-zero weight.

W	$\lvert\mu(T)\rvert$	$2W+3$	p	$\lvert A(p)\rvert$	$\lvert\{a(n)\}\rvert$	$\lvert\{b(n)\}\rvert$
1	4	4	5	11	3	5
2	14	6	8	50	12	22
3	42	8	11	138	27	54
4	132	10	14	308	56	121
5	429	12	17	590	96	200
6	1 430	14	20	1 035	161	337
7	4 862	16	23	1 682	243	482
8	16 796	18	26	2 600	364	709
9	58 786	20	29	3 837	510	936
10	208 012	22	32			
11	742 900	25	35			

The elementary step in our earlier enumerations was the addition of a single new site and two new bonds. On the triangular lattice it is convenient to divide the basic step as shown in figure 5, firstly adding up to two new bonds in the three possible positions and then assigning them labels and secondly linking the bonds at the new site and determining the label of this new site.

The various possible results are shown in tables 2 and 3. The types of the bond that can be added depends on both the type and position of the site to which the bond is added. The possible cases are listed in table 2 which also shows that type 3 sites cannot have bonds added. The other constraint is that at most two of the three new bonds added to the three preceding sites are allowed to be occupied. Table 3 shows the way in which the label of the site in positions 'a' and 'c' changes with the addition of new bonds. After the second part of the basic step, site 'b' is no longer on the transect line, being replaced by the new site.

The type of the new site is determined by the rules shown in table 4. There are two aspects to this table. The first is the type (0, 1, 2 or 3) of the new site; the second is the way in which the resulting state vector is manipulated. The various possibilities are:

Table 2. Changes in site type due to adding bonds from sites in position a and c, for various values of the original site index. Bonds cannot be extended from sites of type 3.

Old site value	0	1	2
Position a	2	3	3
Position c	1	3	3

Table 3. Bond type for bonds added to sites of various types in positions a, b and c. Bonds cannot be added to sites of type 3 nor to site b if its type is 0.

Old site value	0	1	2
Position a	1	1	2
Position b	*	1	2
Position c	2	1	2

(i) *New vector*: Accumulate the product $x^m \mu_\nu(T)$ into the running total for the new vector component $\mu_{\nu'}(T')$ with ν' derived from the labels for ν with the states of sites 'a' and 'c' replaced by the values taken from table 3 and the state of site 'b' replaced by the state of the new site, determined from table 4 using bond states that are determined from table 2.

(ii) *Accumulate*: Accumulate $b(n)x^m\mu_\nu(T)$ into the running total for the triangular lattice polygon generating function where $b(n)$ is the weighting factor determined according to section 3 below.

(iii) *Change 1*: (This occurs when two bonds of type '2' are connected.) Construct a new label set as described in (i) and then transform it by working through decreasing site positions (starting at the new site) and transforming the first unmatched '1' to a '2'.

(iv) *Change 2*: (This occurs when two bonds of type '1' are connected.) As for (ii) but work through increasing site positions changing the first unmatched '2' to a '1'.

The topological basis for these transformation rules is illustrated by Enting (1980).

The construction is initialized by setting column -1, the column immediately outside the left of the finite lattice, so that the configuration with all sites in state 3 has weight 1 and all other configurations have weight 0. These rules entail a change from the approach used in previous studies. Previously we required the polygons to span the length of the finite lattice. The present rules simply require the polygon to be in contact with the right-hand end of the finite lattice.

The elementary step shown in figure 5 adds two triangular faces to the finite lattice. It is straightforward to determine whether one or both of these faces lies inside the partly completed polygon. This involves looking at the whole of the vector index describing the bond intersections and not merely the local bond states. Once this is done an extra variable y can be included to give a y^{area} dependence for enumerating polygons by both area and perimeter. The changes to the algorithm are minor but the costs in computation and storage involved in carrying the additional information about areas restricts the length to which we can derive such two-variable series.

Table 4. State of new site for various combinations of bonds. The triples $[x, y, z]$ denote the states of bonds from the a, b and c sites respectively, with 0 denoting the absence of a bond. The $\rightarrow$ denotes accumulating the left-hand side into the running total of the right-hand side. In the second row, $b(n)$ is the weighting factor for the current hexagon. 'Change 1' and 'Change 2' refer to transformation of the index as described in the text. Once this is done, and in the other cases, the index η is derived form ν by using the new site 'b' as specified in the second column to replace the old site 'b' and using new values for sites 'a' and 'c' as specified in table 3.

Bond states	Site state	Operation
[0,0,0]	0	$\mu_\nu(T) \rightarrow \mu_\eta(T')$
[1,2,0], [1,0,2], [0,1,2]	3	$b(n)x^2\mu_\nu(T) \rightarrow C(x)$
[2,1,0], [2,0,1], [0,2,1]	3	$x^2\mu_\nu(T) \rightarrow \mu_\eta(T')$
[1,1,0], [1,0,1], [0,1,1]	3	Change 2, $x^2\mu_n u \rightarrow \mu_\eta(T')$
[2,2,0], [2,0,2], [0,2,2]	3	Change 1, $x^2\mu_n u \rightarrow \mu_\eta(T')$
[1,0,0], [0,1,0], [0,0,1]	1	$x\mu_n u \rightarrow \mu_\eta(T')$
[2,0,0], [0,2,0], [0,0,2]	2	$x\mu_n u \rightarrow \mu_\eta(T')$

The complexity of the calculations depends on the width (i.e. the maximum width) of the finite lattice. Previous studies have used rectangular lattices so that the calculation uses a sequence of fixed widths. The combinatorial relations presented in the following section require hexagonal lattices so that the width will vary as the lattice is constructed. The rules presented here are entirely consistent with the possibility of varying widths of lattice; configurations with bonds lying outside the lattice are given a weight of zero. An important difference from the square lattice calculations is that for a given width, a partially constructed lattice may be the immediate predecessor of more than one hexagonal lattice. Thus it is desirable to store some of the state vectors for future use. In constrast, if only rectangles are used, there is a unique successor at each step and so there is no need to ever store state vectors after they have been used once.

3. Combining finite lattice results

3.1. Rectangular lattice symmetry

As in previous studies, we use g_{mn} to denote the number of polygons that fit into a rectangle of width n and length m but not into any smaller rectangle. On the triangular lattice, the requirement that the polygons touch the right-hand end of the rectangle means that for any width W and length L we are enumerating the quantities

$$H_{WL} = \sum_{m=1}^{L} \sum_{n=1}^{W} (W - n + 1)\, g_{mn}. \tag{3.1}$$

In contrast, in the earlier square lattice calculations (where the polygons were forced to span the lengths of the rectangles) it was the quantities that were enumerated:

$$G_{WL} = \sum_{n=1}^{W} (W - n + 1)\, g_{mn}. \tag{3.2}$$

These relations hold regardless of whether g_{mn} refers to square lattice polygons or triangular lattice polygons. Thus on any lattice

$$G_{WL} = H_{WL} - H_{W,L-1}. \tag{3.3}$$

The inversion relation

$$g_{mn} = G_{mn} - 2G_{m-1,n} + G_{m-2,n} \tag{3.4}$$

used in the square lattice case gives

$$g_{mn} = H_{mn} - 2H_{m-1,n} + H_{m-2,n} - H_{m,n-1} + 2H_{m-1,n-1} - H_{m-2,n-1}. \tag{3.5}$$

We write this as the linear combination

$$g_{mn} = \sum_{m' \leqslant m, n' \leqslant n} a_{mnm'n'} H_{m'n'}. \tag{3.6}$$

To enumerate polygons we require the sum

$$C(x) \approx \sum_{m,n:n+n \leqslant P, m \leqslant n} b_{mn} g_{mn} \tag{3.7a}$$

with

$$b_{mm} = 1 \qquad \text{and} \qquad b_{mn} = 2 \qquad \text{for} \qquad m < n. \tag{3.7b}$$

Thus the truncated expansion of $C(x)$ will be expressed as a linear combination of H_{mn} for $m + n \leqslant P$ for some maximum semi-perimeter P. Figure 6 shows the smallest triangular lattice polygons not fitting in any rectangle of perimeter $\leqslant 2P$. There are two of them and they have $P + 2$ steps. Thus expansions using rectangles of width $\leqslant W$ include (after use of rotational symmetry on the square lattice) all rectangles of semi-perimeter $\leqslant 2W + 1$ and so will have their first incorrect term at order $2W + 3$. Thus, as indicated in table 1, the series is correct to order $2W + 2$. However for each W, at order $2W + 3$ only the two polygons shown in figure 6 are missing and so one extra series term can be obtained by adding a correction of 2 to the first incorrect term.

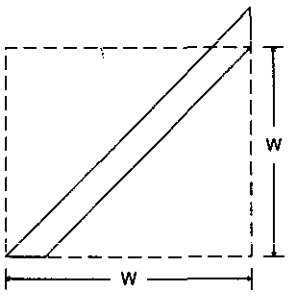

Figure 6. The smallest polygon that fails to fit into a rectangle of width W. This determines that the number of terms that are given correctly by the finite lattice method using rectangular symmetry will be $2W + 2$. However the polygon shown occurs in only six ways, four of which do fit within rectangles of width W and so the correction at order $2W + 3$ is always 2 and can be added in by hand.

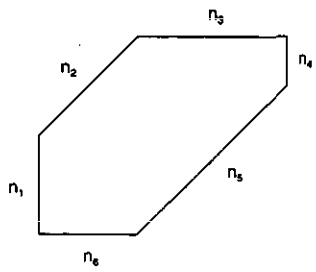

Figure 7. Notation used in describing convex hexagons in terms of the lengths of the sides n_1 to n_6.

A test version of our computer program, using the combinatorics of rectangular lattice symmetry described here, was produced as one step in the development of the more powerful program implementing the formalism described in the following subsection. As reported by Enting and Guttmann (1990), this test version enabled us to extend the known triangular lattice polygon series to 24 steps (or 25 with the correction). Running the program with various maximum widths, W, confirmed that the correction was indeed 2 at each stage.

3.2. Triangular lattice symmetry

From figure 6, it is clear that the missing graphs that limit the series are quite narrow. They can readily be enumerated using transfer matrix techniques if the full triangular lattice symmetry can be exploited. Enting (1987) has described the combinatorics of the finite lattice method on the triangular lattice for Potts model expansions. We now generalize this work to the problem of enumerating polygons.

The enumeration requires that, in principle, we must combine generating functions for polygons embedded in all convex hexagons up to some maximum size. (The derivation below represents a constructive proof of this assertion.) We use the notation of Enting (1987) and denote a general convex hexagon by the sextuple $\boldsymbol{n} = [n_1, n_2, n_3, n_4, n_5, n_6]$, where, as shown in figure 7, n_i is the number of bonds along side i.

We also define the breadth in each of three directions by

$$b_1 = n_1 + n_1 = n_4 + n_5 \tag{3.8a}$$

$$b_2 = n_2 + n_3 = n_5 + n_6 \tag{3.8b}$$

$$b_3 = n_3 + n_4 = n_6 + n_1. \tag{3.8c}$$

These equations give two independent constraints on the possible sets of n_i. We have to include degenerate cases where some of the n_i are zero. For polygon enumeration we can ignore cases in which any of the b_i are zero although such cases are required when applying the formalism of Enting (1987) to low-temperature expansions.

We define a set $A(p_{\max})$ of hexagons whose perimeter is $p = \sum n_i \leqslant p_{\max}$. For any polygon, α, of perimeter $p(\alpha) \leqslant p_{\max}$ there will be a minimal bounding hexagon $\boldsymbol{n}(\alpha)$ which will have perimeter $p(\boldsymbol{n}) \leqslant p_{\max}$. We let $g(\boldsymbol{n})$ denote the generating function for all polygons whose minimal bounding rectangle is $\boldsymbol{n}$, i.e.

$$g(\boldsymbol{n}) = \sum_{\alpha:\boldsymbol{n}(\alpha)=\boldsymbol{n}} x^{p(\alpha)} \tag{3.9}$$

and so

$$C(x) \approx \sum_{\boldsymbol{n} \in A(p)} g(\boldsymbol{n}) \qquad \text{correct to } x^p. \tag{3.10}$$

The direct enumeration of the $g(\boldsymbol{n})$ by transfer matrix techniques is not particularly convenient and so for $\boldsymbol{n} \in A(p_{\max})$ we enumerate $h(\boldsymbol{n})$ which is the generating function for polygons that can be embedded in hexagon $\boldsymbol{n}$ with at least one point in contact with side n_4. Thus

$$h(\boldsymbol{m}) = \sum_{\boldsymbol{n} \subseteq \boldsymbol{m}} X(\boldsymbol{n}, \boldsymbol{m})\, g(\boldsymbol{n}) \tag{3.11}$$

where $X(n,m)$ is the number of ways n can be embedded in m with the line segmanet n_4 lying within line segment m_4. In particular $X(n,n) = 1$. In order to use the triangular lattice symmetry to perform the calculations as efficiently as possible, we follow Enting (1987) and define a canonical representation for each of the classes of hexagons that are equivalent under the symmetry operations. The canonical representative of a class has the width $n_1 + n_2$ chosen to be as small as possible; in the case of equality the minimum n_2 is chosen and then the minimum n_5 and finally the minimum n_6. If we let $a(n)$ be the number of hexagons equivalent to n when n is canonical and let $a(n) = 0$ otherwise then the polygon generating function becomes

$$C(x) \approx \sum_{n \in A(p)} a(n)\, g(n) \qquad \text{correct to } x^p. \tag{3.12}$$

In order to write this in the form

$$C(x) \approx \sum_{m \in A(p)} b(m)\, h(m) = \sum_{m \in A(p)} \sum_{n \subseteq m} b(m)\, X(n,m)\, g(n) \tag{3.13}$$

we need the coefficients $b(m)$ which satisfy

$$\sum_{m \in A(p_{\max})} X(n,m)\, b(m) = a(n) \qquad \text{for } n \in A(p_{\max}). \tag{3.14}$$

Since $X(n,m)$ is an upper triangular matrix, its inverse will also be an upper triangular matrix and so the solution will be of the form.

$$b(m) = \sum_{n \supseteq m} Y(m,n)\, a(n). \tag{3.15}$$

Thus if, by definition, the $a(n)$ are non-zero only for canonical hexagons then the $b(m)$ will be non-zero only for canonical hexagons and their subgraphs. (A complete proof of this assertion involves making use of the flexibility that is possible when constructing a total ordering of matrix indices from the partial ordering defined by the subgraph relation.) In table 1 we give, for each width, the number of convex hexagons of perimeter $p \leqslant 3W + 2$, the number of canonical hexagons in this set and the number of hexagons having non-zero weight. These numbers differ from those given by Enting (1987) because hexagons of breadth 0 have been excluded.

As noted by Enting (1987) the matrix structure allows us to solve for the $b(n)$ by constructing one column of the matrix at a time and eliminating according to the following procedure:

For all m in decreasing order:

(i) construct the column $X(n,m)$; and

(ii) perform the elimination $a(n) \leftarrow a(n) - X(n,m)a(m)$ for all $n \subset m$.

The final value of $a(n)$ is the required solution $b(n)$. The elements $X(n,m)$ are constructed as described by Enting (1987) with the simplification that only translations in one direction must be considered, because of the constraint that the '4' edges must overlap.

Once the combinatorial factors, $b(n)$, have been obtained, the transfer matrix procedure must be modified to determine the generating functions $h(n)$ for hexagons n. The procedure was:

(1) Loop over all breadths $n_1 + n_2 \leqslant W$.
(2) For each breadth, loop over all neccessary values of n_2.
(3) Initialize a finite lattice.
(4) For each n_6 up to the maximum required, add a column to the finite lattice, adjusting the upper boundary as specified by side n_2.
(5) For each n_6, if non-zero values of n_5 are needed, a duplicate copy of the partially completed generating function is stored for future use and the transfer matrix procedure progresses, adding n_5 more columns, and adjusting the lower boundary to follow side n_5. (The upper boundary continues to be adjusted while $n_6 + n_5 \leqslant n_2$.)
(6) When all n_5 values for the particular combintion of n_1, n_2, n_6 have been processed, the stored generating function is used to restart the procedure at step 4 for the next n_6 value.

Compared with the earlier calculation, the use of triangular lattice symmetry involves looping over n_2 and n_5 in addition to the loops over breadth and length that occur in each case. In addition the storage requirements when using triangular symmetry are greater because of the need to store a vector of partly constructed generating function to restart the procedure after branching to loop over n_5. For these reasons going to width 11 required significantly more compting resources than the width 11 calculation using rectangular symmetry. Going beyond width 11 would have required even more resources and so we have been limited to width 11, thereby enumerating polygons for $p \leqslant 35$. Retaining the information about areas imposes an additional cost in both storage and computation (the factor is proportional to the maximum area considered) and so our enumerations by area and perimeter have used only width 8. The results of these calculations are given in tables 5 and 6.

4. Analysis of the series

The calculations required a working array of about 75 Mbyte. They were run on an IBM 6000/530 computer with 64 Mbyte of memory. As the array was randomly accessed, the execution time was largely dominated by disc transfer rates. As a result, the execution time was 30 days. In order to minimize storage, the calculations were carried out in 2-byte integer arithmetic. Two distinct prime integers close to and smaller than 2^{15} were chosen, and the calculations were carried out *modulo* each prime. The results were reconstructed, giving the answers *modulo* the product of the two primes, approximately 10^9. This gave the nine least significant digits in the polygon count, the most significant digits being obtained from differential approximants, as described in Guttmann and Enting (1988). Generating the coefficients of the two-variable area-perimeter generating function, we counted all polygons with area $\leqslant 24$ triangles. The maximum perimeter of such polygons is 26 steps. Thus we obtained the early terms in both the polygon generating function

$$P(x) = \sum p_n x^n \tag{4.1}$$

and the two-variable generating function

$$P(x, y) = \sum p_{m,n} x^m y^n \tag{4.2}$$

where p_n is the number of n-step polygons with perimeter n, and $p_{m,n}$ is the number of polygons with perimeter m and 'area' n. To ensure that both m and n are integers,

Table 5. Enumeration of triangular lattice polygons. For n, x_n is the number of triangular lattice polygons with area of n unit triangles and c_n is the number with perimeter n.

n	x_n	c_n	n	c_n
1	2	—	25	233 893 503 330
2	3	—	26	880 918 093 866
3	6	2	27	3 329 949 535 934
4	14	3	28	12 630 175 810 968
5	36	6	29	48 056 019 569 718
6	94	15	30	183 383 553 173 255
7	250	42	31	701 719 913 717 994
8	675	123	32	2 692 047 018 699 717
9	1 832	380	33	10 352 576 717 684 506
10	5 005	1 212	34	39 902 392 511 347 329
11	13 746	3 966	35	154 126 451 419 554 156
12	37 901	13 265		
13	104 902	45 144		
14	291 312	155 955		
15	811 346	545 690		
16	2 265 905	1 930 635		
17	6 343 854	6 897 210		
18	17 801 383	24 852 576		
19	50 057 400	90 237 582		
20	141 034 248	329 896 569		
21	398 070 362	1 213 528 736		
22	1 125 426 581	4 489 041 219		
23	3 186 954 777	16 690 581 534		
24	9 039 483 589	62 346 895 571		

we adopt the metric that the lattice has unit lattice spacing, which ensures that m is an integer, and that the area of an elementary triangle is also unity, rather than the correct Euclidean value of $\sqrt{3}/4$.

The series have been analysed by the methods employed previously in our study of square and honeycomb lattice polygons, and detailed in Guttmann (1987). To save space, we will not tabulate the various approximants as we did in our earlier papers. Suffice it to say that the results are quite comparable in apparent accuracy and rate of convergence. Our analysis used both first- and second-order differential approximants. Writing the polygon generating function as

$$P(x) \sim A(x)(1-\mu x)^{2-\alpha} \tag{4.3}$$

the estimates obtained for the critical point and critical exponent may be summarized as:

$1/\mu = 0.2409175 \pm 0.0000003$ $\quad \alpha = 0.49991 \pm 0.0002$ (first order)

$1/\mu = 0.2409174 \pm 0.0000003$ $\quad \alpha = 0.49979 \pm 0.0004$ (second order).

These results are comparable in accuracy to those obtained for the honeycomb lattice. Linear regression, coupled with the assumption $\alpha = \frac{1}{2}$ exactly, yields the biased estimate

$$1/\mu = 0.2409177 \pm 0.0000002.$$

Table 6. Numbers of triangular lattice polygons $p_{m,n}$ grouped by area, m, and perimeter, n.

n	m	$p_{m,n}$	n	m	$p_{m,n}$	n	m	$p_{m,n}$	n	m	$p_{m,n}$
3	1	2	12	10	4 603	15	13	90 128	19	17	4 992 132
4	2	3	12	12	4 460	15	15	134 336	19	19	10 837 134
5	3	6	12	14	2 631	15	17	124 992	19	21	14 291 136
6	4	14	12	16	1 113	15	19	90 926	19	23	14 843 706
6	6	1	12	18	356	15	21	55 202	20	18	13 711 572
7	5	36	12	20	87	15	23	28 992	20	20	32 086 809
7	7	6	12	22	14	16	14	244 755	20	22	45 278 676
8	6	93	12	24	1	16	16	406 608	20	24	50 012 364
8	8	27	13	11	12 360	16	18	419 913	21	19	37 746 064
8	10	3	13	13	14 136	16	20	341 796	21	21	94 713 162
9	7	244	13	15	10 032	16	22	235 662	21	23	142 597 599
9	9	110	13	17	5 340	16	24	142 737	22	20	104 123 868
9	11	24	13	19	2 244	17	15	666 780	22	22	278 835 717
9	13	2	13	21	774	17	17	1 221 348	22	24	448 067 007
10	8	648	13	25	48	17	19	1 381 026	23	21	287 775 834
10	10	399	14	12	33 306	17	21	1 234 254	23	23	819 091 488
10	12	135	14	14	43 899	17	23	943 710	24	22	796 743 323
10	14	27	14	16	36 153	18	16	1 822 028	24	24	2 401 443 942
10	16	3	14	18	22 893	18	18	3 646 649	25	23	2 209 449 072
11	9	1 722	14	22	5 187	18	20	4 469 850	26	24	6 136 210 872
11	11	1 362	14	24	1 917	18	22	4 328 002			
11	13	636				18	24	3 604 749			
11	15	198									
11	17	42									
11	19	6									

In order to estimate the correction-to-scaling exponent, we constructed the new generating function $P(x)(1-\mu x)^{-3/2}$, using the biased estimate of μ. Analysis of that generating function indicated a singularity at $x = 1/\mu$ with an exponent of $-\frac{3}{2}$ (a divergence). This implies the presence of an analytic correction term to (4.3), so that

$$P(x) \sim A(x)(1-\mu x)^{2-\alpha} + B(x). \tag{4.4}$$

We have also repeated the analysis reported in Enting and Guttmann (1989), in which we use the method of Baker and Hunter (1973) for the analysis of confluent exponents, which requires an accurate estimate of the critical point. There was no evidence of a confluent singularity. As with the other lattices, we therefore find that the correction-to-scaling exponent is as predicted by Nienhuis (1982, 1984), and is $\Delta = 1.5$. This is, therefore, indistinguishable from the analytic background term. We do, however, find it remarkable that there is no evidence of a term with exponent $3-\alpha$, which would follow from the expansion of $A(x)$ in (4.3), unless $A'(1/\mu) = 0$.

We have also estimated the amplitude, $A(1/\mu)$ by dividing out by the singular part and evaluating the resulting series at $x = 1/\mu$ by both sequence extrapolation and differential approximant methods. In this way we estimate A(triangle) = 0.6239, A(square) = 1.3289, A(honeycomb) = 3.0053. Another amplitude that is frequently encountered is defined by

$$p_n = B\mu^n n^{-5/2}[1 + o(n)].$$

Clearly, $B = A/\Gamma(-3/2)$. From these amplitudes, and amplitudes that are calculable from previous work, we present in table 7 a summary of amplitudes for a variety of quantities that arise in the two-dimensional SAW (self-avoiding walk) problem. In all, we give results for seven distinct amplitudes on the three common two-dimensional lattices. In some cases the series is not available to estimate the required quantity, but in all such cases scaling or conformal invariance arguments allow the relevant quantity to be estimated. The quantities considered are:

(i) the chain generating function $C(x) = \sum c_n x^n$,
(ii) the polygon generating function defined by (4.1),
(iii) the mean-square end-to-end distance of an n-step SAW $\langle R_n^2 \rangle$,
(iv) the mean-square radius of gyration of n-step polygons, $\langle R_g^2 \rangle_p$,
(v) the mean area of polygons of perimeter n, $\langle a_n \rangle$,
(vi) the mean-square radius of gyration of n-step SAWs $\langle R_g^2 \rangle_w$; and
(vii) the mean-square distance of a monomer from the origin of an n-step SAW, $\langle R_m^2 \rangle$.

The relevant amplitudes are defined as follows.

$$c_n = A\mu^n n^{\gamma-1}[1 + o(1)]$$

$$p_n = B\mu^n n^{\alpha-3}[1 + o(1)] \qquad \text{(even terms only for loose-packed lattices)}$$

$$\langle R_n^2 \rangle = Cn^{2\nu}[1 + o(1)]$$

$$\langle R_g^2 \rangle_p = Dn^{2\nu}[1 + o(1)] \qquad \text{(even terms only for loose-packed lattices)}$$

$$\langle a_n \rangle = En^{2\nu}[1 + o(1)] \qquad \text{(even terms only for loose-packed lattices)}$$

$$\langle R_g^2 \rangle_w = Fn^{2\nu}[1 + o(1)]$$

$$\langle R_m^2 \rangle = Gn^{2\nu}[1 + o(1)]$$

where $\gamma = 43/32$, $\alpha = 1/2$, $\nu = 3/4$ and $\mu = 1/x_c$. For the honeycomb lattice $\mu = (2 + \sqrt{2})^{1/2}$, for the square lattice our best estimate is $\mu = 2.6381585$ (Guttmann and Enting 1988), and for the triangular lattice we have from the result above $\mu = 4.150795$.

Table 7. Estimates of amplitudes A, B, C, D, E, F and G respectively the amplitudes of the coefficients of the chain generating function, the polygon generating function, the mean-square end-to-end distance of SAWs, the mean-square radius of gyration of polygons, the mean area of polygons of a given perimeter, the mean-square radius of gyration of SAWs, the mean-square distance of a monomer from the origin in a SAW. Precise definitions are given in the text. Quantities estimated from scaling relations are shown parenthesized.

Amplitude	Honeycomb	Square	Triangular
A	1.145	1.178	1.186
B	1.272	0.5623	0.264
C	(0.884)	0.770	0.711
D	(0.0249)	0.0563	(0.120)
E	(0.0625)	0.1416	0.302
F	(0.124)	0.108	0.0997
G	(0.389)	0.339	0.313

For the chain generating function we have 42 (unpublished), 34 (Masand *et al* 1992) and 22 terms (Guttmann and Wang 1991) respectively in the series expansion for the honeycomb, square and triangular lattices. For the polygon generating function we have the series to $n = 82$ (Enting and Guttmann 1989), 56 (Guttmann and Enting 1988) and 35 (this paper) for the honeycomb, square and triangular lattices respectively. For $\langle R_n^2 \rangle$ we have the series to order 29 and 22 for the square and triangular lattices (Guttmann and Wang 1991), and for $\langle R_g^2 \rangle$ we have a 28 term series on the square lattice (Privman and Rudnick 1985). For $\langle a_n \rangle$ we have the series up to $n = 42$ and 13 for the square (Enting and Guttmann 1990) and triangular lattices (shown later) respectively. For $\langle R_g^2 \rangle_w$ and $\langle R_m^2 \rangle$ we have unpublished series data to 28 terms on the square lattice and 19 terms on the triangular lattice, extending by several terms the earlier work of Guttmann and Yang (1990).

From this data we have estimated the amplitudes A, B, C, D, E, F and G for all lattices for which data were available. Several methods of analysis were used, based on extrapolation algorithms and differential approximants (Guttmann 1989). The results are summarized in table 7. Any uncertainty is limited to one or two in the last quoted digit. Uncertainties in the critical points are too small to affect these estimates.

There are several scaling relations that allow the table to be filled. Cardy (1988) has shown that $BD = 5/(16\pi^2)$. This gives D for the honeycomb and triangular lattices, from the estimate of B. Privman and Redner (1985) have shown that $CB/(v\sigma)$ is a universal quantity, where $\sigma = 1$ for a close-packed lattice and $\sigma = 2$ for a loose-packed lattice. v measures the unit-cell size, and is 1 (square) $\sqrt{3}/2$ (triangular) and $3\sqrt{3}/2$ (honeycomb). From our estimates we find this quantity to be 0.2168 (triangular) and 0.2165 (square). This allows C(honeycomb) to be estimated. Camacho and Fisher (1990) have shown that E/D is universal, and Fisher *et al* (1991) have estimated this quantity to be 2.511 ± 0.001. This allows E(honeycomb) to be estimated, and allows a second estimate of D(triangular), which agrees precisely with the estimate obtained from Cardy's relation. From conformal invariance theory, Cardy and Saleur (1989) showed that F/C and G/C are lattice independent for two-dimensional lattices, and also showed (after subsequent correction by Caracciolo *et al* 1990) that

$$\left(2 + \frac{y_t}{y_n}\right)\frac{F}{C} - \frac{2G}{C} + \frac{1}{2} = 0$$

where $y_t = 4/3$ and $y_h = 91/48$ are the known thermal and magnetic eigenvalues respectively for the two-dimensional SAW problem. These scaling relations permit F and G for the honeycomb lattice to be estimated. This completes table 7.

These various relationships between the amplitudes raises the question whether more constant ratios could be identified. Sokal (private communication) points out that both F/C and G/C could be rational given the fact that their ratio is rational. Cardy (private communication) points out that F/C and the mean-square area could, in principle, be evaluated. This follows from Zamolodchikov's exact S-matrix for the $O(n)$ model. An impressive amount of algebra would be required to carry out these calculations however.

Turning now to the two-variable generating function, we have repeated the analysis we carried out for the square lattice in Enting and Guttmann (1990) and Fisher *et al* (1991). The area generating function, $P(1, y)$ is found to diverge with exponent

zero (presumably corresponding to a logarithm or some power thereof) at $y_c = 0.3394 \pm 0.0003$. Hence

$$P(1, y) \sim G(y) + H(y) \log(1 - y/y_c). \tag{4.5}$$

A range of rigorous results was presented for the square lattice version of this problem by Fisher *et al* (1991), and these all hold *mutatis mutandis* in the triangular case. In particular, $P(x, y)$ converges for $y < 1$ only when $x < x_c(y)$, and for $y > 1$ only for $x = 1$. The form of the phase boundary is qualitatively the same as for the square lattice. The arguments in Fisher *et al* can be repeated here with but slight modification. Note however that the minimum area $n_{\min}$ for fixed perimeter m is $n_{\min} = m - 2$ (with the metric that the lattice has unit lattice spacing and that the area of an elementary triangle is also unity). It follows that $x_c(y) \sim c/y$ as $y \to 0$, compared to the corresponding result $x_c(y) \sim c/y^{1/2}$ for the square lattice. (With a Euclidean metric the corresponding result would be $x_c(y) \sim c/y^{\sqrt{3}/4}$). From the data in table 6 we have also calculated the mean area series,

$$n_m(1) = \sum n p_{m,n} / \sum p_{m,n} = N_0 m^{2\nu}[1 + o(1)] \tag{4.6}$$

where $\nu = 3/4$ and the amplitude N_0 is found to be 0.1416 ± 0.0003, as given in table 7.

5. Conclusions

We have shown how the finite lattice method for enumerating self-avoiding polygons may be extended to the triangular lattice, and used the method to greatly extend the existing polygon series. Indeed we have increased the number of polygons counted by a factor in excess of 10^{10}. We also present for the first time the coefficients of the two-variable generating function, giving polygons by both area and perimeter. Analysis of various series has permitted precise estimates of the critical points to be made, has permitted numerical confirmation of Nienhuis' (1982, 1984) estimate of the critical exponent α, and combined with previous work and various universal quantities, has allowed a complete tabulation of various critical amplitudes for self-avoiding walks and rings the three common two-dimensional lattices. We find no evidence of a non-analytic correction-to-scaling exponent from the polygon generating function.

Acknowledgments

We have benefited from discussions with John Cardy, Bernard Nienhuis and Alan Sokal. We would like to thank IBM (Australia) for the loan of memory for an IBM 6000/Model 530 on which these calculations were performed. Financial support from the Australian Research Council is also acknowledged.

References

Baker G A Jr and Hunter D L 1973 *Phys. Rev.* B 7 3377
Camacho C J and Fisher M E 1990 *Phys. Rev. Lett.* 65 9

Caracciolo S, Pelissetto A and Sokal A D 1990 *J. Phys. A: Math. Gen.* **23** L969
Cardy J L 1988 *J. Phys. A: Math. Gen.* **21** L797
Cardy J L and Saleur H 1989 *J. Phys. A: Math. Gen.* **22** L601
Enting I G 1980 *J. Phys. A: Math. Gen.* **13** 3713
—— 1987 *J. Phys. A: Math. Gen.* **20** 1585
Enting I G and Guttmann A J 1985 *J. Phys. A: Math. Gen.* **18** 1007
—— 1989 *J. Phys. A: Math. Gen.* **22** 1371–84
—— 1990 *J. Stat. Phys* **58** 475
Enting I G and Wu F Y 1982 *J. Stat. Phys.* **28** 351
Fisher M E, Guttmann A J and Whittington S 1991 *J. Phys. A: Math. Gen.* **24** 3095
Guttmann A J 1987 *J. Phys. A: Math. Gen.* **20** 1839
—— 1989 *Phase Transitions and Critical Phenomena* vol 13, ed C Domb and J Lebowitz, (New York: Academic).
Guttmann A J and Enting I G 1988 *J. Phys. A: Math. Gen.* **21** L165
Guttmann A J and Wang J 1991 *J. Phys. A: Math. Gen.* **24** 3107
Guttmann A J and Yang Y S 1990 *J. Phys. A: Math. Gen.* **23** L117
Masand B, Wilensky U, Massar J P and Redner S 1991 *J. Phys. A: Math. Gen.* **25** L365
Nienhuis B 1982 *Phys. Rev. Lett.* **49** 1062
—— 1984 *J. Stat. Phys.* **34** 731
Privman V and Redner S 1985 *J. Phys. A: Math. Gen.* **18** L781–8.
Privman V and Rudnick J 1985 *J. Phys. A: Math. Gen.* **18** L789
Sykes M F, McKenzie D S, Watts M G, and Martin J L 1972 *J. Phys. A: Math. Gen.* **5** 661