Self-avoiding paths on a generalized cayley tree

J. Phys. A: Math. Gen., Vol. 8, No. 5, 1975. Printed in Great Britain. © 1975

Self-avoiding paths on a generalized Cayley tree

R Haydock and W B Temple

Cavendish Laboratory, Madingley Road, Cambridge CB3 0HE, UK

Received 27 September 1974, in final form 27 November 1974

Abstract. Self-avoiding paths and unrestricted or free paths are investigated on a lattice obtained from a Cayley tree by replacing vertices by clusters of vertices and edges by bundles of edges. Closed analytic expressions are obtained for the generating functions, and the statistics of paths up to length 25 are explored.

1. Introduction

The physics and chemistry of polymeric systems is a field rich in unsolved problems. Since the models used to describe such systems are still rudimentary, we find that many of the problems encountered are combinatorial in nature, leading naturally to the application of graph theory and related techniques (Gordon and Temple 1972). Such problems can usually be stated in a few words although their solution may present formidable mathematical difficulty. Typical of this class is the 'excluded volume' problem in polymer physics.

A simple model for a linear polymer chain is the 'pearl necklace' model where the monomers are the beads, each joined to its neighbours by a rigid link. The chain has extensive flexibility by virtue of bond rotation, and under the influence of random thermal motion this entity will assume some 'random coil' configuration. If we suppose that the chain does not interact with itself, we may generally extract the relevant statistical parameters characterizing the spatial domain occupied by the random coil by using Gaussian flight statistics (Flory 1969). We now impose the condition that the chain must self-avoid, that is no two beads may occupy the same volume of space. Attempts to calculate exactly the effect on the random coil have met with rather limited success, although the self-consistent field treatment of a continuum model by Edwards is outstanding in this field (Edwards 1965). Rigorous analytical results, therefore, are few but there is now a substantial accumulation of evidence suggesting that distributions become non-Gaussian (Domb 1969).

For reasons touched on below, the modelling of a polymer chain configuration with excluded volume by a self-avoiding walk or path on a crystal lattice has received much attention. From work by Hammersley (1957, 1961) and Kesten (1964) we have two important asymptotic analytical results:

$$C_n \sim f_1(n)\mu^n \tag{1}$$

$$u_n \sim f_2(n)\mu^n. \tag{2}$$

Here C_n is the total number of self-avoiding paths (SAP) of n steps, u_n is the total number of SAP which return to a nearest neighbour of the origin after n steps, μ is a constant, usually called the connective constant, which is lattice-dependent and may be thought of as a sort of coordination number, and $f_1(n)$ and $f_2(n)$ are functions such that

$$\lim_{n\to\infty} (f_1(n))^{1/2} = \lim_{n\to\infty} (f_2(n))^{1/2} = 1.$$

Monte Carlo techniques have been used, principally by Wall *et al* (1963), to generate samples of long walks ($n \simeq$ 100–500), a method which entails high rates of attrition. A method which relates more closely to the work described here has been developed largely by Domb and collaborators (1969) and involves performing exhaustive, exact enumerations of short paths (up to about 15 for 3D lattices). From the behaviour of these short paths, extrapolation for asymptotic properties may be carried out with reference to equations (1) and (2). Both the above methods rely on ingenious strategy and efficient programming to keep the computing times involved within reasonable bounds.

Here we present a method for computing SAP with small computing effort, using generating functions of closed form. However, this is achieved only by the use of a non-physical lattice.

2. The method

We could investigate the properties of self-avoiding paths in a number of different situations. For their relation to polymers the best way would be to define them as simple curves in three-space such that two points whose separation along the curve is greater than some distance r never come closer than r. This idealization is too complicated to solve, as is the simplification that we consider SAP on lattices in three-space rather than all curves. The argument is that although the properties of the SAP on lattices depend on the lattice, there is basic SAP behaviour which is independent of the lattice. The use of lattices does not simplify the basic problem which is that of an interaction between all pairs of points on the path; it does make the SAP denumerable and thus open to numerical investigation on a computer.

There is a further simplification of the problem which makes the interaction simpler and allows an analytic solution. That is to construct a lattice on which a SAP can only interact with itself if it fulfils certain conditions. On a real lattice a SAP can curve back on itself from anywhere, but on some lattices this is not possible. One such lattice is a Cayley Tree or Bethe Lattice. The tree consists of vertices each with a number of branches such that only the closed paths retrace themselves. A tree for $z = 4$ is shown in figure 1. The Cayley tree has proved very useful for investigating paths on lattices with the same coordination number z (Brinkman and Rice 1970). However, it is clear that the SAP on the Cayley tree are very simple and uninteresting, namely that there is exactly one between each pair of vertices.

We regain the interesting SAP structure by generalizing the Cayley tree, in that each vertex is considered as a cluster of vertices connected together and each edge between vertices of the old tree as a bundle of edges between the clusters. Figure 2 shows the result of these substitutions on five vertices of the Cayley tree in figure 1. In figure 2 there are four tetrahedrally placed vertices in each cluster and that they are a generalization of one vertex is indicated by the symbol at each new vertex. The vertices marked by circles all came from one old vertex. The vertices marked by circles with a cross came from another and so forth. Each face of the tetrahedron of circles has three

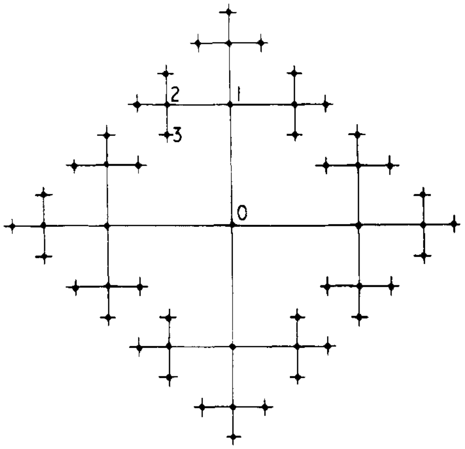

Figure 1. A portion of a four-coordinated Cayley tree.

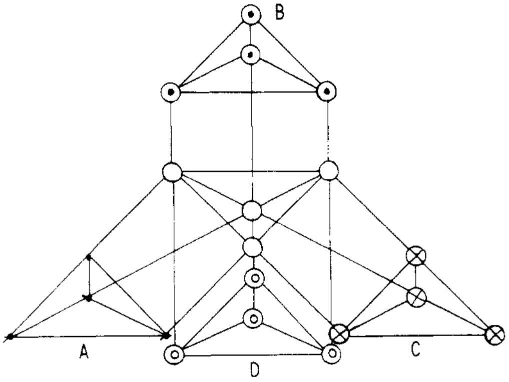

Figure 2. A portion of a generalized Cayley tree.

edges connecting it to a face of a neighbouring tetrahedral cluster. As soon as we introduce this complexity into the vertices and edges of the tree the SAP become interesting. But the interaction takes place only for paths that retrace their way through the same clusters. This advantage makes the generating function for the SAP obtainable analytically.

There are several objections to the generalized Cayley tree (GCT). The first is the same one as for use of the Cayley tree, namely that the lattice cannot fit in three-space and thus the results bear no relation to real lattices. However, in many problems the Cayley tree results are very similar to real lattice results. The reason for this is that locally the Cayley tree is very similar to a real lattice with the same z. The GCT is locally even more like the real lattice in that it contains clusters right out of the real lattice.

The second objection is that the GCT is not angularly uniform. A path which enters a cluster needs different numbers of steps to reach neighbouring clusters, depending on the direction of the neighbouring cluster. This can be seen in figure 2 from the fact that it is shorter from A to B than to a similar point C. We feel that this problem is not great, and the results support this view.

The GCT in figure 2 is one we have used for this investigation of the SAP. Our approach is to define $G_{AB}(S)$, the generating function for SAP between vertices A and B, as

$$G_{AB}(S) = \sum_{n=0}^{\infty} \mu_n S^n \tag{3}$$

where μ_n is the number of SAP between A and B of length n. First we consider the case where A and B are both on the same tetrahedron (figure 3). If a SAP from A to B involves steps outside the cluster, the SAP must depart and return on one of the four groups of three edges which connect the cluster to other clusters. In order to account for these SAP, we introduce another generating function $g(S)$ for the SAP which connect two vertices of the tetrahedron by an excursion into nearby clusters:

$$g(S) = \sum_{n=0}^{\infty} \mu_n' S^n \tag{4}$$

where μ_n' is the number of SAP of length n connecting two vertices of a face by an excursion on the branch of the GCT connected to that face. Note that each pair of vertices belongs to two faces and so can be connected by SAP excursions on two branches of the GCT.

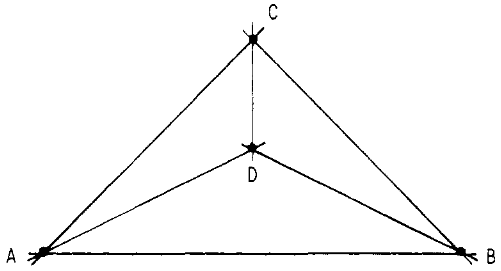

Figure 3. A cluster in the generalized Cayley tree.

If a path leaves from a particular face, it must return to that face by the Cayley tree property of the GCT. A SAP can make at most one excursion out of a face because in doing so it uses up two of the three vertices on that face.

Now by adding up the various SAP in the tetrahedron accounting for excursions out of it by $g(S)$, we can obtain an expression for $G_{AB}(S)$:

$$G_{AB}(S) = S+2S^2+2S^3+2(1+4S+6S^2)g(S)+2(3+10S)g(S)^2+8g(S)^3. \tag{5}$$

Now by a similar method we can obtain an equation for $g(S)$. Figure 4 shows a tetrahedron which connects to that in figure 3. We use this to obtain an equation for $g(S)$. Excursions from cluster ABCD can go out of any of the four faces to a cluster like EFGH. For example, suppose the SAP exits from face ACD along AE, then in exactly the same way as it could leave ABCD by four faces it can leave EFGH by any of the three faces other than EFG. We can now write equations for $g(S)$ analogous to $G_{AB}(S)$ by counting up the SAP in EFGH and using $g(S)$ for excursions further out. The same restrictions apply as before and we obtain

$$g(S) = S^3+2S^4+2S^5+(S^2+6S^3+10S^4-1)g(S)+(4S^2+12S^3)g(S)^2+2S^2g(S)^3. \tag{6}$$

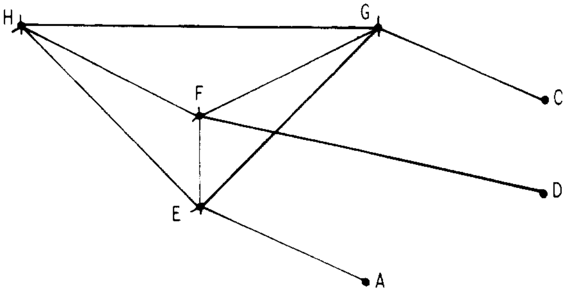

Figure 4. Connection between clusters in the generalized Cayley tree.

This equation is cubic in $g(S)$ so there are three roots. However, only one root tends to unity as S goes to zero, so there is no ambiguity. Once we have this solution, $G_{AB}(S)$ follows by substitution in (5).

$G_{AB}(S)$ gives the number of SAP between two vertices on the same cluster. However we would like to find the number of SAP joining vertices in different clusters. The problem is thus to enumerate the different ways in which a SAP can connect clusters. Referring to figure 1, but thinking of it now as a GCT with the vertices representing clusters and the edges representing bundles of edges, if the endpoints of a SAP are in different clusters there is only one route on the GCT between them, that is the SAP must pass through all the clusters between the endpoints at least once. The complication is that the SAP may pass through a cluster in our model up to three times. The method is: for each cluster on the direct route between endpoints we consider all ways by which the path can pass through the cluster.

Figure 3 shows a cluster which we take to be on the route such that the initial vertex lies off the ABC face and the final vertex lies off the DBC face. There is a total of nine ways by which a SAP can enter or exit from a face. These ways are enumerated in table 1. For each pair of input and output states, m and l, of the cluster, there is a generating function $g_{ml}(S)$, where the coefficient of S^n is the number of SAP links between the entrance and exit. The intercluster links are counted on the entrance side. This count includes excursions from the other two faces, ACD and ABD. In this formulation we have allowed the SAP to double back on itself, which means that there may be up to three disconnected links of the SAP in a cluster. The generating functions are given in table 2. A similar set of generating functions applies should the route leave either ACD or ABD.

Table 1. Classification of different ways by which a SAP can enter or leave the tetrahedron in figure 3.

State	Entering ABC	Exiting DBC
1	in A	out D
2	in B	out B
3	in C	out C
4	in A then out B then in C	out D then in B then out C
5	in A then out C then in B	out D then in C then out B
6	in B then out A then in C	out B then in D then out C
7	in B then out C then in A	out B then in C then out D
8	in C then out A then in B	out C then in D then out B
9	in C then out B then in A	out C then in B then out D

Table 2. The generating functions $g_{lm}(S)$.

l, m	g_{lm}
1, 1	$S^2+2Sg+2S^3+2S^4+2S^2g^2+4S^2g+4S^3g$
1, 2	$S^2+Sg+4S^2g+2S^3+2S^4+Sg^2+6S^3g+3S^2g^2$
1, 3	g_{12}
1, 4	S^3+2S^2g
1, 5	g_{14}
1, 6	$g_{14}+Sg^2$
1, 7	g_{16}
1, 8	g_{16}
1, 9	g_{16}
2, 1	g_{12}
2, 2	S
2, 3	$S^2+2S^3+2Sg^2+2S^4+4S^2g+8S^3g+6S^2g^2$
2, 6	$S^2+Sg+S^3+3S^2g+Sg^2$
2, 7	g_{26}
3, 1	g_{13}
3, 2	S
3, 8	g_{26}
3, 9	g_{26}
4, 1	$g_{14}\times S^2$
4, 4	S^4+2S^3g
5, 5	g_{44}
6, 6	g_{44}
7, 7	g_{44}
8, 8	g_{44}
9, 9	g_{44}
All l, m not listed above	$\equiv 0$

At the initial (or final) cluster there is yet another generating function $\alpha(\mathrm{S})$ (or $\Omega(S)$) which links A with the incoming state for the second cluster (or B with the outgoing state for the penultimate cluster) in the chain of clusters separating the origin and terminus of the path. Therefore, these generating functions count paths just as above, but are designed to take care of special conditions in the initial and final cluster.

The $\alpha(S)$ (or $\Omega(S)$) are constructed in exactly the same way as the other generating functions, by summing the contributions for all SAP which start at A (or finish at B) and go into state m as the SAP leaves (or enters) that cluster. The generating functions for all SAP from A to B in different clusters are thus given by

$$G_{\mathrm{AB}}(S) = \sum_{m_1 m_2 \dots m_n} \alpha g m_1 m_2 \dots g m_{n-1} m_n \Omega. \tag{7}$$

As well as the SAP we have considered the free paths (FP) in this model in order to compare the two. The approach for FP is similar to that for SAP. We again derive generating functions for the FP connecting various vertices in a cluster. Let us define (referring to figure 3) $\tilde{G}_{\mathrm{AB}}(S)$ as the generating function for FP between vertices A and B. We introduce two new generating functions $\tilde{f}(S)$ and $\tilde{g}(S)$. $\tilde{f}(S)$ accounts for all excursions from a particular face (say ABC) returning to the same vertex on that face. Therefore, $\tilde{f}(S)$ generates the FP which start an excursion into neighbouring clusters from A, say, and which first rejoin the face ABC through vertex A. $\tilde{g}(S)$ is the FP analogue of the SAP $g(S)$ and generates the number of FP which execute the excursion described above, but which first rejoin the face ABC via another vertex on that face, say B. We can now construct a matrix of links between vertices α, β:

$$\tilde{H}_1(S) = \begin{pmatrix} 3\tilde{f}(S) & S+2\tilde{g}(S) & S+2\tilde{g}(S) & S+2\tilde{g}(S) \\ S+2\tilde{g}(S) & 3\tilde{f}(S) & S+2\tilde{g}(S) & S+2\tilde{g}(S) \\ S+2\tilde{g}(S) & S+2\tilde{g}(S) & 3\tilde{f}(S) & S+2\tilde{g}(S) \\ S+2\tilde{g}(S) & S+2\tilde{g}(S) & S+2\tilde{g}(S) & 3\tilde{f}(S) \end{pmatrix} \tag{8}$$

where the columns and rows correspond to the different vertices in a cluster. Each matrix element corresponds to the sum of those internal and external FP which connect two vertices on the cluster, which touch these two vertices only once (at the beginning and end) and which do not visit either of the remaining two vertices. From the matrix of direct connections of vertices in the cluster we can construct the FP which touch A, B, C or D an arbitrary number of times by considering the powers of $\tilde{H}_1(S)$. In fact the matrix of generating functions $\tilde{G}(S)$ for the FP between points in ABCD is just the sum of powers of $\tilde{H}_1(S)$. Formally,

$$\tilde{G}_1(S) = (I - \tilde{H}_1(S))^{-1}. \tag{9}$$

Turning to the evaluation of $\tilde{f}(S)$ and $\tilde{g}(S)$, similar considerations apply as in equation (6). Considering excursions out of face ACD into figure 4 and beyond, a FP which only touches A, C or D at the beginning and end can only cross EA, FD or GC at the beginning and end, so we could consider the link matrix for EFGH and exclude excursions out of the face ACD. Doing this yields the following matrix:

$$\tilde{H}_2(S) = \begin{pmatrix} 2\tilde{f}(S) & S+\tilde{g}(S) & S+\tilde{g}(S) & S+2\tilde{g}(S) \\ S+\tilde{g}(S) & 2\tilde{f}(S) & S+\tilde{g}(S) & S+2\tilde{g}(S) \\ S+\tilde{g}(S) & S+\tilde{g}(S) & 2\tilde{f}(S) & S+2\tilde{g}(S) \\ S+2\tilde{g}(S) & S+2\tilde{g}(S) & S+2\tilde{g}(S) & 3\tilde{f}(S) \end{pmatrix} \tag{10}$$

where the columns and rows correspond to the vertices of EFGH. Just as for $\tilde{G}_1(S)$ we obtain a generating function $\tilde{G}_2(S)$ for all FP between EFGH by generating all powers of $\mathrm{H}_2(S)$:

$$\tilde{G}_2(S) = (I - \tilde{H}_2(S))^{-1}. \tag{11}$$

From the definition of $\tilde{f}(S)$ and $\tilde{g}(S)$ they must be respectively S^2 times the EE and EF elements of $\tilde{G}_2(S)$. This gives a pair of coupled polynomial equations for $\tilde{f}(S)$ and $\tilde{g}(S)$. The solution of these is substituted back into equation (9) to obtain $\tilde{G}_1$ for the first cluster. Again there is no ambiguity about solutions because $\tilde{f}(S)$ and $\tilde{g}(S)$ must go to unity as S tends to zero. This eliminates all but one pair of solutions.

Having obtained the generating function for FP between vertices in the same cluster, we move on to consideration of FP between vertices in different clusters. In this case we analyse the route rather differently. The FPs can double back on themselves an arbitrary number of times, but we can break an FP down into sections such that, for each successive face on a tetrahedral cluster along the main route, we divide the FP into two so that the later section never comes back through that plane. Notice that in adopting this procedure we overlook no paths, as it is clear that every path must make a final exit from a cluster on the way to its endpoint, after having passed through it an arbitrary number of times beforehand. Hence we can construct all FP between two points by multiplying together the generating functions for FP which begin and end in each successive face along the route and which are not allowed to touch vertices further back along the route. Between these generating functions we sandwich a factor of S to account for the intercluster links. For example, if our route were 0123 in figure 1 our first factor would be a vector of generating functions which take us from the starting vertex to the exit face of cluster 0, followed by a factor of S to take us to the entrance face of cluster 1. Next is a matrix of generating functions to take us from the entrance face to the exit face of cluster 1 without going out of 1 towards 0; then a factor of S to take us to 2, and another entrance-to-exit matrix excluding excursions towards 1, and so on. We finish with another vector which takes us from the entrance face of 3 to the final vertex excluding excursions back towards 2.

The generating vector for the first vertex to the first face is simply the appropriate elements of $\tilde{G}_1(S)$. The face-to-face matrix is constructed from the appropriate elements of $\tilde{G}_2(S)$ because returning along the route is forbidden, and the final vector comes again from $\tilde{G}_2(S)$. For instance, the generating function for FP from B in figure 3 to H in figure 4 is expressed as

$$\tilde{G}_{\mathrm{BH}}(S) = (\tilde{G}_{1\mathrm{BA}}(S), \tilde{G}_{1\mathrm{BC}}(S), \tilde{G}_{1\mathrm{BD}}(S))S \begin{pmatrix} \tilde{G}_{2\mathrm{EH}}(S) \\ \tilde{G}_{2\mathrm{GH}}(S) \\ \tilde{G}_{2\mathrm{FH}}(S) \end{pmatrix}. \tag{12}$$

We have now defined the generating functions for all SAP and FP in our GCT. We wish to obtain from these the numbers of various kinds of paths between various points. This involves first solving for the generating function and then evaluating the coefficients of various powers of S in it. The method we used for this is based on Cauchy's integral theorem and involved complex integrations (Delves and Lyness 1967, Nex 1967).

Although these methods are well known, we will outline them here. Given a pair of coupled polynomial equations in unknowns u and v,

$$P_1(u, v, S) = 0 \qquad P_2(u, v, S) = 0, \tag{13}$$

if there is in a neighbourhood of $S = 0$ a unique $u(S)$ and $v(S)$, each of which goes to unity as S tends to zero, then we can draw contours C_u and C_v around the corresponding neighbourhoods in u and v, so that for a given S:

$$\begin{gathered} u(S) = \frac{1}{(2\pi \mathrm{i})^2} \oint_{Cu} \mathrm{d}u \oint_{Cv} \mathrm{d}v \frac{u}{P_1}\left(\frac{\partial}{\partial u} P_1\right) \frac{1}{P_2}\left(\frac{\partial}{\partial v} P_2\right) \\ \text{and} \\ v(S) = \frac{1}{(2\pi \mathrm{i})^2} \oint_{Cv} \mathrm{d}v \oint_{Cu} \mathrm{d}u \frac{v}{P_2}\left(\frac{\partial}{\partial v} P_2\right) \frac{1}{P_1}\left(\frac{\partial}{\partial u} P_1\right). \end{gathered} \tag{14}$$

These formulae follow from Cauchy's theorem and the uniqueness of u and v in the neighbourhoods. This uniqueness can be checked by counting the number of roots, N, of the equations in the neighbourhoods:

$$N = \frac{1}{(2\pi \mathrm{i})^2} \oint_{Cu} \mathrm{d}u \oint_{Cv} \mathrm{d}v \frac{1}{P_1}\left(\frac{\partial}{\partial u} P_1\right) \frac{1}{P_2}\left(\frac{\partial}{\partial v} P_2\right). \tag{15}$$

Once the coupled equations have been solved the generating functions come from the multiplication of appropriate terms. The number of paths between A and Z of length n, which is the coefficient μ_n of S^n in $G_{AZ}(S)$, also comes from Cauchy's theorem. Let C_S be a contour around the neighbourhood of $S = 0$ where u and v are unique; then

$$\mu_n = \frac{1}{2\pi i}\oint_{CS} \mathrm{d}S\, S^{-n-1} G_{AZ}(S)\, \mathrm{d}S. \qquad (16)$$

In this numerical work errors can arise in two ways. For relatively nearby points, as n increases in equation (16) S^{-n-1} oscillates more and more along C_S, which means that more points must be taken in the integration. The second source of error comes in paths between distant points. The generating functions for these paths are constructed by multiplication of many matrices, in which rounding error becomes important.

3. Results

A preliminary result, which gives an illustration of the effect of the self-avoiding potential on paths of length 25 links is given in figure 5. The contrast in the probability distributions for the separation of endpoints of FP (line A) and SAP is apparent. The paths were generated on the following basis, which may be described with reference to figure 6. The

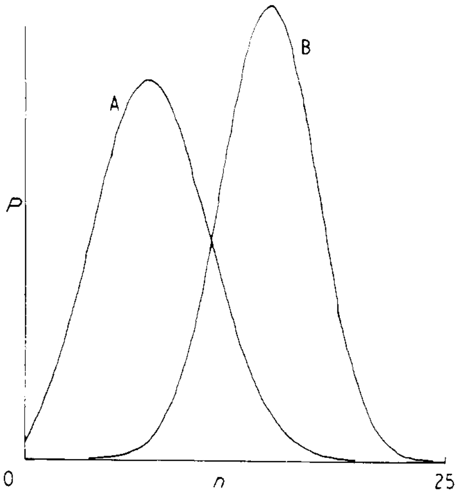

Figure 5. The probability P of finding the endpoint of a path of length 25 on the shell of tetrahedra which are $n-1$ tetrahedra away from the starting tetrahedron. Curve A is for free paths and curve B is for self-avoiding paths.

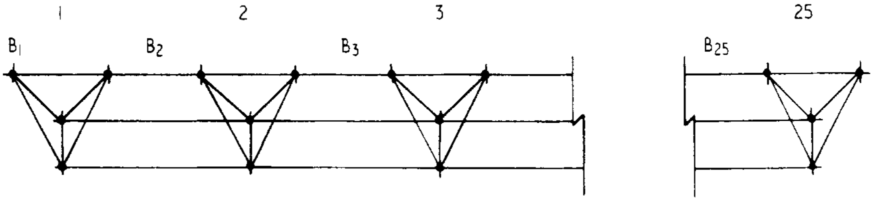

Figure 6. Numbering system for paths between points in different tetrahedra.

starting point of the path was taken as vertex A on tetrahedron 1 (the root tetrahedron). As mentioned above the lattice is not isotropic, so for the distribution plots of figure 5 we enumerated paths which had their termini at vertices $B_1, B_2, B_3, \ldots, B_{25}$ on the string of tetrahedra connected as shown in figure 6. This particular chain of clusters is of course contained in the lattice, and we believe that it furnishes a fairly typical statistical subset of paths. For the SAP the appropriate matrix of generating functions is given in table 2, with special generating function vectors (α, Ω) for initial and endpoint tetrahedra given in table 3.

Table 3. Initial (α) and endpoint (Ω) vectorial generating functions for the enumeration of SAP in figure 5.

	α	Ω	
α_1	$S+g+2S^2+2S^3+2g^3+6Sg+4g^2+10S^2g+12Sg^2$	Ω_1	S
α_3	α_1	Ω_2	$\alpha_1 \times S$
		Ω_3	Ω_2
α_6	$S+g+S^2+4Sg+3g^2$	Ω_7	$\alpha_6 \times S^3$
α_7	α_6	Ω_9	Ω_7
	all other $\alpha_i \equiv 0$	all other $\Omega_i \equiv 0$	

The numerical results were obtained as outlined in § 2 by multiplying generating function matrices and vectors and by performing the requisite complex integrations on the computer. The FP enumerations were carried out in similar fashion. Table 4 gives the answers obtained from the computer, together with an (pessimistic) error estimate from the integration routine. As n increases each endpoint B_n finds itself on a tetrahedron further away from the root tetrahedron. In the GCT there are $4 \times 3^{n-2}$ 'similar' vertices in the tetrahedron shell corresponding to B_n. Hence for each B_n in table 4 the relevant number of FP or SAP was weighted by multiplication by $4 \times 3^{n-2}$, giving when normalized the probability curves of figure 5.

Notice that when paths of length 25 change from free to self-avoiding in character, the most probable endpoint position moves from tetrahedron 8 to tetrahedron 15. There is also a very marked effect on those paths which return to the neighbourhood of the origin, these being severely attenuated for SAP. The problem of ring formation in polymer systems is of course heavily dependent on return-to-the-origin statistics. Usually fewer rings are detected experimentally than are predicted by theories which neglect excluded volume. Although we believe the GCT approach to overstate this depletion near the origin (since a SAP is constrained to return to the root tetrahedron *only* via the face from which it left it), the analogous effect in the proper 3D lattice (where returns to the origin are permitted from any direction) would lead to depressed cyclization of polymer chains.

The GCT treatment thus produces results which are qualitatively interesting, but we must enquire how such a non-physical lattice compares with respectable 3D crystal lattices. We turn to equations (1) and (2) which have been widely used to interpret data from a variety of real lattices. It has been fairly well established that $f_1(n)$ may be represented by n^g. Substituting in equation (1) we find

$$C_n/C_{n-1} \sim \mu(1+g/n). \qquad (19)$$

Table 4. The computed numbers of FP and SAP between the endpoints shown in figure 6. The paths are of length 25 and all start at vertex A.

Endpoints	No. of SAP of length 25	Error estimate	No. of FP of length 25	Error estimate
B_1			$5{\cdot}9776554 \times 10^{16}$	8×10^{13}
B_2	$1{\cdot}8502862 \times 10^{10}$	7×10^{-3}	$4{\cdot}8920049$	4×10^{11}
B_3	$3{\cdot}1991095$	6×10^{-5}	$3{\cdot}0882918$	1
B_4	$4{\cdot}5463671$	6	$1{\cdot}6617577$	9×10^{9}
B_5	$5{\cdot}6726674$	4	$7{\cdot}8910301 \times 10^{15}$	2
B_6	$6{\cdot}2828878$	3	$3{\cdot}3589208$	2×10^{7}
B_7	$6{\cdot}1292753$	5	$1{\cdot}2939573$	1
B_8	$5{\cdot}4261620$	3	$4{\cdot}5371561 \times 10^{14}$	2×10^{6}
B_9	$4{\cdot}2125498$	1	$1{\cdot}4543108$	1×10^{5}
B_{10}	$2{\cdot}8860712$	2	$4{\cdot}2713537 \times 10^{13}$	2×10^{4}
B_{11}	$1{\cdot}7382478$	2	$1{\cdot}1518952$	8×10^{2}
B_{12}	$9{\cdot}1771460 \times 10^{9}$	2	$2{\cdot}8532871 \times 10^{12}$	2
B_{13}	$4{\cdot}2404206$	6×10^{-6}	$6{\cdot}4960113 \times 10^{11}$	3×10^{1}
B_{14}	$1{\cdot}7140112$	3	$1{\cdot}3571107$	1×10^{0}
B_{15}	$6{\cdot}0609782 \times 10^{8}$	3	$2{\cdot}6001587 \times 10^{10}$	1×10^{-1}
B_{16}	$1{\cdot}8745865$	2	$4{\cdot}5493403 \times 10^{9}$	2×10^{-2}
B_{17}	$5{\cdot}0648179 \times 10^{7}$	1	$7{\cdot}2558203 \times 10^{8}$	3×10^{-3}
B_{18}	$1{\cdot}1917825$	1	$1{\cdot}0455775$	4×10^{-4}
B_{19}	$2{\cdot}4283740 \times 10^{6}$	8×10^{-7}	$1{\cdot}3574886 \times 10^{7}$	2×10^{-5}
B_{20}	$4{\cdot}2434300 \times 10^{5}$	6	$1{\cdot}5581040 \times 10^{6}$	2×10^{-7}
B_{21}	$6{\cdot}2601000 \times 10^{4}$	5	$1{\cdot}5771800 \times 10^{5}$	3
B_{22}	$7{\cdot}5990000 \times 10^{3}$	4	$1{\cdot}3429000 \times 10^{4}$	3×10^{-8}
B_{23}	$7{\cdot}2600000 \times 10^{2}$	3	$9{\cdot}6800000 \times 10^{2}$	4×10^{-9}
B_{24}	$5{\cdot}0000000 \times 10^{1}$	2	$5{\cdot}0000000 \times 10^{1}$	2×10^{-10}
B_{25}	$2{\cdot}0000000 \times 10^{0}$	2	$2{\cdot}0000000 \times 10^{0}$	1×10^{-11}

Evidence suggests that g is lattice-independent and approximates to $\frac{1}{6}$ for 3D lattices (Domb 1969).

By permuting the rows of the generating function matrices and vectors and by forming composite matrices and vectors, it was possible to modify the SAP enumeration program described above to generate C_n values for our GCT. That is we were able to generate all SAP of length up to 25, not only for those which had their endpoints on the tetrahedron chain of figure 6. Figure 7 is a plot of C_n/C_{n-1} against $1/n$. According to

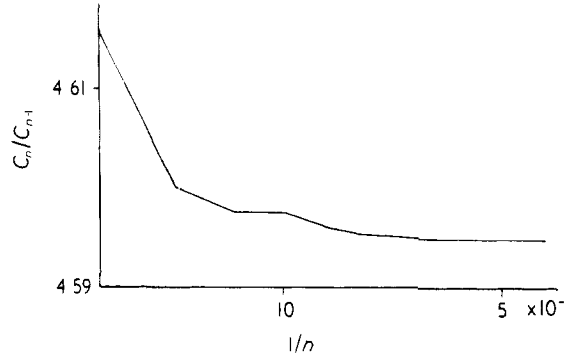

Figure 7. The ratios of numbers of SAP of lengths n and $n-1$.

equation (19) this plot should be linear with a slope yielding g and an intercept equal to μ. In this case asymptotic behaviour sets in around $n = 13$ and extrapolation gives $\mu = 4{\cdot}59475$, with uncertainty only in the last figure. For real lattices good asymptotic behaviour is achieved at smaller values of $n\,(n \sim 7)$. A simple cubic lattice (which has the same coordination number of 6 as our lattice) gives $\mu = 4{\cdot}68260 \pm 0{\cdot}00025$ (Sykes 1963) which compares reasonably with our value. Our value of g, however, is about 9×10^{-5}—very much smaller than $\frac{1}{6}$. Indeed the slope seems to be tending asymptotically to zero. This we may ascribe to the decoupled nature of the GCT lattice and is reinforced by the small cluster size of 4 vertices. This is brought out more clearly in a plot of U_n/U_{n-1} against $1/n$ shown in figure 8. $f_2(n)$ seems to have the same form for real lattices as $f_1(n)$ except that g now approximates to $-\frac{7}{4}$ (Domb 1969). That is plots tend to be asymptotically linear and to slope upwards (with reference to the C_n/C_{n-1} plot) to intercept the vertical axis at the same value of μ. In our lattice this is manifestly. not the case. The plot seems to have a persistent curvature, even though we have enumerated paths of length up to 75 (for $n = 75$ the number of SAP returning to a nearest-neighbour vertex to the origin was computed as $4{\cdot}160878563 \times 10^{34}$ with an error estimate of 2×10^{24}). It is reasonable that the assumptions of a GCT lattice should be specially significant in the region of the origin, since a path can only return to the root tetrahedron via the face from which it last left it.

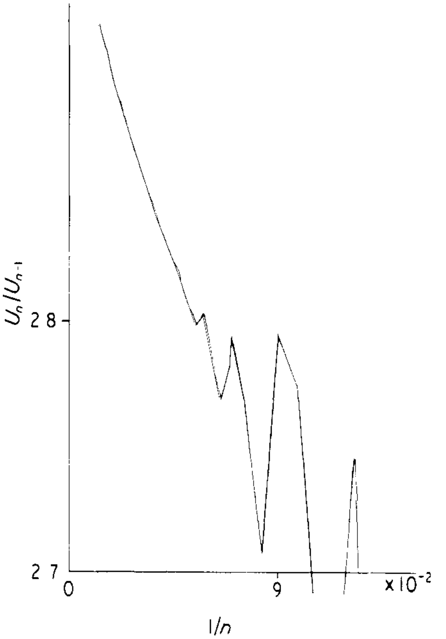

Figure 8. The ratios of numbers of SAP of lengths n and $n-1$ which return to a nearest neighbour of the starting vertex.

4. Conclusions

These preliminary results show that using the GCT approach, well defined generating functions and reliable computational techniques may be set up to solve enumeration problems on a structure which admits excluded volume effects. Careful choice of contours for complex integrations and the use of multiple precision arithmetic in the computer will extend the range of path lengths which can be examined far beyond that reported here. The method may also prove helpful in related topics in the areas of percolation theory (Domb and Sykes 1961), Ising type problems (Domb 1960) and theories of cooperative phenomena (Rushbrooke and Eve 1962). If the method could be extended for larger clusters (eg cubes instead of tetrahedra), we should expect the results to correspond more nearly to reality.

However, the comparison with real 3D lattices indicates that in making the simplifications implicit in the GCT approach we may well have lost an essential feature of the problem we wish to study.

Acknowledgments

The authors would like to acknowledge the support of the Drapers' Research Fellowship at Pembroke College, Cambridge for RH and of a Science Research Council Research Grant for WBT.

References

Brinkman W F and Rice T M 1970 *Phys. Rev.* B **2** 1324–38
Delves L M and Lyness J N 1967 *Math. Comp.* **21** 543–77
Domb C 1960 *Adv. Phys.* **9** 149, 245–361
—— 1969 *Adv. Chem. Phys.* **15** 229–59
Domb C and Sykes M F 1961 *Phys. Rev.* **122** 77–8
Edwards S F 1965 *Proc. Phys. Soc.* **85** 613–24
Flory P J 1969 *Statistical Mechanics of Chain Molecules* (New York: Wiley Interscience)
Gordon M and Temple W B 1973 *J. Chem. Soc.* **69** 282–97
Hammersley J M 1957 *Proc. Camb. Phil. Soc.* **53** 642–5
—— 1961 *Q. J. Math., Oxford* **12** 250–6
Kesten H 1964 *J. Math. Phys.* **5** 1128–37
Nex C M M 1967 *MSc Dissertation* University of Sussex
Rushbrooke G S and Eve J 1962 *J. Math. Phys.* **3** 185–9
Sykes M F 1963 *J. Chem. Phys.* **39** 410–2
Wall F T, Windwer S and Gans P J 1963 *Methods in Computational Physics* vol 1 (New York: Academic)