Self-Avoiding Hamiltonian Walks Counting in Parallel Processing Mode

Igor Ševo[1], Sreten Lekić[2], and Mihajlo Savić[1]

[1] Faculty of Electrical Engineering, University of Banja Luka,
Bosnia and Herzegovina
[2] Faculty of Natural Sciences and Mathematics, University of Banja Luka,
Bosnia and Herzegovina
badaboom@etfbl.net

Abstract. We have developed a program for counting self-avoiding Hamiltonian walks to run on multiple processors in a parallel mode. We study Hamiltonian walks (HWs) on the family of two-dimensional modified Sierpinski gasket fractals, as a simple model for compact polymers in nonhomogeneous media in two dimensions. We apply an exact recursive method which allows for explicit enumeration of extremely long Hamiltonian walks of different types: closed and open, with end-points anywhere in the lattice, or with one or both ends fixed at the corner sites. The leading term n is characterized by the value of the connectivity constant 1, which depends on fractal type, but not on the type of HW.

Keywords: hamiltonian walks, fractal, parallel processing.

1 Introduction

Self-avoiding walks (SAWs) have been used in the modeling of polymers, micro magnetics configuration, studies of diffusion through the porous media, statistics of polymer chains configuration in solution [1]. The most intensive preoccupations are directed toward the polymer phase transitions. Due to excluded volume effect, at high temperatures T (good solvent) long polymer chains are in swollen configurations. At low temperatures (poor solvent) polymers are in a collapsed state, caused by the attractive interactions of different sections of a polymer, mediated by a solvent. The transition between these two states occurs at the θ temperature, at which excluded volume and attractive forces balance. Whereas the swollen and θ phases has been well investigated by now, the entropy scaling of the collapsed phase is still an open issue. A closely related problem is the scaling of Hamiltonian walks (HWs), which are SAWs that visit all the sites of the underlying lattice [2]. HWs represent the $T = 0$ limit of collapsed polymers, and they are also used in the studies of polymer melting, as well as in the context of protein folding [3]. The number Z_N of HWs on homogeneous lattices for large N behaves as $Z_N \sim \omega^N \mu^{N_S^\sigma} N^\alpha$ where $\sigma = (d-1)/d$, (d is the dimensionality of the lattice), μ is constant less than 1, and ω is the connectivity

DOI: 10.1007/978-3-319-01520-0_7, © Springer International Publishing Switzerland 2014

constant. Proposed scaling form for HWs differs from the ordinary SAW case (swollen polymer), where average number of N-step SAWs, for large N, behaves as $\omega^N N^\alpha$, and critical exponent α depends only on d (which is not the case for HWs). The term $\mu^{N_S^\sigma}$ is result of the exact study of HWs on the Manhattan lattice [4], and the consequence of conjecture that collapsed polymer (globule) has a sharp boundary and surface tension terms are more dominant [5].

2 Hamiltonian Walks on Sierpinski Gasket Fractals

SG is a well known fractal lattice, which can be constructed recursively, starting with the generator (gasket of order $r = 1$), which consists of three unit equilateral triangles, arranged to form a larger triangle (see Fig. 1). The subsequent fractal stages are constructed self-similarly, by replacing each of the unit triangles of the initial generator with a new generator. To obtain the rth-stage fractal lattice (rth order gasket), this process of construction has to be repeated $(r-1)$ times, and the complete fractal is obtained in the limit $r \to \infty$ numbers of sites on the rth order gasket is equal to $N_r = \frac{3}{2}(3^r + 1)$. SG resembles 3-simplex lattice and indeed has the same fractal dimension $df = \ln 3/\ln 2$. An open HW on a third order gasket is shown on Fig. 3, together with its coarse-grained versions. Comparing with Fig. 1 one can observe that larger number of types of possible HW configurations exists on SG than in the case of 3-simplex lattice. There are exactly eight different types of walks, and they are depicted on Given-Mandelbrot fractals represent the fractal family characterized with integer $b \geq 2$ (scaling factor) [7]. The first element of row is Sierpinski gasket, fractal with $b = 2$. At the same way, other fractal are constructed k. Initiator is *a* site triangle *a*. Aa the first step of construction, $r = 1$, $\frac{b(b+1)}{2}$ initiator sets in equilateral triangle ba on this way that the vertices are connected.

Constructed structure represents the generator of order one $G^{(1)}(b)$. In the second steep, $r = 2$, $\frac{b(b+1)}{2}$ initiator set on the same way in equilateral triangle b^2a site. This is the generator of order 2. After r iterative steps generator order r, $G^{(r)}(b)$, triangle with $b^r a$ site has $\frac{b(b+1)}{2}$ generators $(r-1)$ order, and complete fractal when $r \to \infty$. The first 3 generators of GM fractal $b = 3$ parameter is on fig 1.

Fractal dimension is

$$d_f = \frac{\ln \frac{b(b+1)}{2}}{\ln b} \,. \tag{1}$$

Initiator vertexes and sites of GM fractal make fractal lattice. The number of nodes, N_r, of $G^{(r)}(b)$, is recursive expressed as $N_r = \frac{b(b+1)}{2}(N_{r-1} - 3) + \sum_1^{b+1} k$. The first term expresses number of nodes of all $\frac{b(b+1)}{2}$ generators of order $r-1$, with vertices exception. Summa presents the number of generators order $r-1$, and is equal $\sum = (b+1)(b+2)/2$. After simplification N_r expression for

$N_r = \frac{b(b+1)}{2}N_{r-1} - b^2 + 1$, and iteration toward the begin, $N_0 = 3$, and

$$N_r = \frac{b+4}{b+2}\Big(\frac{b(b+1)}{2}\Big)^r + 2\frac{b+1}{b+2} \,. \tag{2}$$

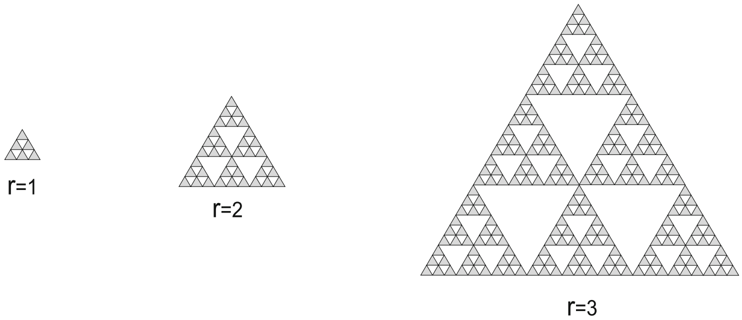

Fig. 1. Construction GM fractal by $b = 3$

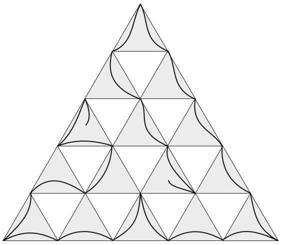

Fig. 2. Open HW on generator $(r + 1)$- order for GM fractal $b = 5$. HW consists od one walk A, one A_1, four B and nine walks B_1 on coarsed generators of order r, and this is the shape $AA_1B^4B_1^9$ in reduced notation.

The corresponding numbers of HWs on the rth-2 order gasket will be denoted by $A^{(r)}$, $A_1^{(r)}$,$A_1^{(r)}$ $B^{(r)}$,$B_1^{(r)}$,$C^{(r)}$, $D^{(r)}$ and $D_1^{(r)}$.

These numbers fulfill the following recursion relations: Since in [6] it was exactly shown that $B(r) = const \cdot \omega^{N_r}$, with $\omega = 12^{1/9}$, one finally obtains the same scaling form as for 3-simplex lattice: $Z_0(r) = \omega^{N_r} N_r^{\gamma}$, with the same value of exponent $\gamma = \ln 16/ \ln 3$. It was also shown in [8] that the overall number Z(r) C of closed HWs on SG lattice scales according to the formula ω^{N_r}, again the same as in the case of 3-simplex. The equality of exponents for these two fractal lattices is in accord with the fact that SAWs on them belong to the same universality class [10]. On the other hand, it is known that exponents for HWs on different 2d Euclidean lattices have different values [11], which is explained to be a consequence of the frustration, induced by the strong constraint that all the sites must be visited.

It is believed that a relevant physical measure of this frustration is the number of contacts per monomer, i.e. vertex pairs which are not adjacent along the HW, but are the nearest neighbors on the lattice. Nevertheless, the number of contacts on 3-simplex is one, whereas it is two on SG, so that one could have expected different values of γ. In order to gain a deeper insight into the problem of universality and frustration of HWs on lattices embedded in 2d space,the asymptotic behavior of HWs on the appropriate generalizations of 3-simplex and SG fractals was analyzed [12].

3 Hamiltonian Walks on Given-Mandelbrot Fractals

One way to construct the SG fractal is to start with a generator that consists of $b(b+1)/2$ unit equilateral triangles, arranged to form a b times larger triangle. Enlarging the generator b times and substituting the smallest triangles with the generator, and then repeating this procedure recursively, one obtains fractal lattice characterized with the integer b. For b = 2, 3, ... the complete so called Given-Mandelbrot (GM) family of fractals is obtained [9]. SG is the first member of this family, with the scaling parameter $b = 2$.

Number of open HWs on the $(r+1)$th stage of any GM fractal construction can be expressed in terms of numbers of 8 HW types within the rt-h order stage, in a similar manner as in the case of SG. Recursion relations for numbers of B, and B1-type walks on two successive stages of fractal construction have the following form

$$B' = pB^{\frac{b(b-1)}{2}+1}B_1^{b-1}, \qquad B_1' = B_1^b \tag{3}$$

as was shown in [6]. whereas the numbers of one-leg configurations: A, A1, A2, and C, satisfy a closed set of recursion relations, which can be put in matrix form (9) where a_{ij} are polynomials in $B(r)$ and $B(r)_1$.

The number $Z(r+1)$ of open HWs on the gasket of order $(r+1)$, can be expressed as (4) where we have suppressed index r on the right-hand side of this relation, and K_{ij} are numbers that depend only on b. Substituting established asymptotical behavior of A_i, C and D_i in the latter expression, one finds that all terms on the right-hand side of equation have the same asymptotical form. The values of γ for are equal to $2 \leq b \leq 8$ are 2.5237. . ., 2.1841 . . ., 2.3411 . . ., 2.2461 . . ., 2.2981 . . ., and 2.2755 . . ., respectively. One should mention here that number of closed HWs asymptotically behaves as Z(r) as was established in [6], where also a closed formula was derived [8].

$$\begin{aligned}\mathcal{Z}_{r+1} = &\sum_{i\leq j=0}^{2} k_{ij} A_{i\,r} A_{j\,r} B_r^{m-2+(i+j)} B_{1\,r}^{n-(i+j)} + C_r \sum_{i=0}^{2} k_i A_{i\,r} B_r^{m-1+i} B_{1\,r}^{n-1-i} \\ &+ k_3 C_r^2 B_r^m B_{1\,r}^{n-2} + \sum_{i=0}^{1} k_i{}' D_{i\,r} B_r^{m+i} B_{1\,r}^{n-1-i} \,,\end{aligned} \tag{4}$$

where $m = b$ and $n = \frac{b(b-1)}{2}$. Term coefficients k_{ij}, k_i ..., are independent of r, doesn't influence on HW's number.

Recurrent relations B i B_1 are

$$B_{r+1} = p\, B_r^m B_{1\,r}^n \,, \quad B_{1\,r+1} = p\, B_r^{m-1} B_{1\,r}^{n+1} \,, \tag{5}$$

and for $A\,, A_1\,, A_2$ and C :

$$
\begin{aligned}
A_{i\,r+1} &= \sum_{j=0}^{2} a_{ij} A_{j\,r} B_r^{m-1-(i-j)} B_{1\,r}^{n+(i-j)} + a_{i3} C_r B_r^{m-i} B_{1\,r}^{n-1+i}\,, \quad i = 0,1,2\,, \\
C_{r+1} &= \sum_{i=0}^{2} c_i A_{i\,r} B_r^{m-2+i} B_{1\,r}^{n+1-i} + c_3 C_r B_r^{m-1} B_{1\,r}^{n}\,.
\end{aligned} \tag{6}
$$

Term coefficients in recurrent relations for A_1 and A_2 can expressed over the coefficients A and B, $a_{10} = a_{00} - p/2\,, a_{11} = a_{01}\,, a_{12} = a_{02} + p/2\,, a_{13} = a_{03}$ i $a_{20} = a_{00} - p\,, a_{21} = a_{01}\,, a_{22} = a_{02} + p\,, a_{23} = a_{03}$, .

The number od B and B_1 are proportional to r. division of this relation give $\frac{B_{1\,r+1}}{B_{r+1}} = \frac{B_{1\,r}}{B_r}$, and $\frac{B_{1\,r}}{B_r} = const = t = \frac{B_{1(1)}}{B_{(1)}}$ (t is the rate of starting number of walks B_1 i B, i and is the function of fractal parameter b). One variable $A_{i\,r}$ can be expressed over other $A_{2\,r} = 2tA_{1\,r} - A_r t^2$, $\forall\ r > 1$. Elimination $B_{1\,r}$ i $A_{2\,r}$ relations in (5) and (6) go in

$$
\begin{aligned}
B_{r+1} &= pt^n B_r^{m+n}\,, \\
A_{r+1} &= B_r^{m+n+1}\left[t^n(a_{00} - a_{02})A_r + t^{n-1}(a_{01} + 2a_{02})A_{1\,r} + t^{n-1}a_{03}C_r\right], \\
A_{1\,r+1} &= B_r^{m+n+1}\left[t^{n+1}(a_{00} - a_{02} - p)A_r + t^n(a_{01} + 2a_{02} + p)A_{1\,r} + t^n a_{13}C_r\right], \\
C_{r+1} &= B_r^{m+n+1}\left[t^{n+1}(c_0 - c_2)A_r + t^n(c_1 + 2c_2)A_{1\,r} + t^n c_3 C_r\right].
\end{aligned} \tag{7}
$$

Iteration of the first of these relations, or go with solution $B_r = c\,c_1^{(m+n)^r}$), and(2), and definitions t, m i n, the result is[1]

$$
B_r = \mathfrak{B}\omega^{N_r}\,, \quad \text{gdje je} \quad \omega = \Big[p\Big]^{k_1}\Big[B_{(1)}\Big]^{k_2}\Big[B_{1(1)}\Big]^{k_3}\,, \quad \text{i} \tag{8}
$$

$$
\begin{pmatrix} x_{r+1} \\ x_{1\,r+1} \\ y_{r+1} \end{pmatrix} = \frac{1}{p}\begin{pmatrix} a_{00} - a_{02} & (a_{01} + 2a_{02})/t & a_{03}/t \\ t(a_{00} - a_{02} - p) & a_{01} + 2a_{02} + p & a_{03} \\ t(c_0 - c_2) & c_1 + 2c_2 & c_3 \end{pmatrix}\begin{pmatrix} x_r \\ x_{1\,r} \\ y_r \end{pmatrix}\,, \tag{9}
$$

4 Implementation and Parallelization

Initial attempts to find the number of walks was based on sequential Fortran code which, while providing correct results, left a lot to be desired both in performance and scalability department. In order to obtain desirable performance and ease the parallelization we opted to forgo brute-force path counting approach and try to find a more elegant solution.

The problem of building all walks on the gasket can be reduced to the problem of constructing each walk using the predefined elementary triangles. Since, given

[1] $\mathfrak{B} = p^{-\frac{2(b+2)}{b(b-1)(b+4)}}\left[B_{(1)}\right]^{\frac{b(b+2)-4}{b(b+4)}}\left[B_{1(1)}\right]^{-\frac{b+2}{b+4}}$.

Table 1. Coefficients p, a_{0i} and c_i in recursion relations for HW B, A i C for the numbers of B, A i C -type, found by direct computer enumeration of all possible corresponding configurations, together with the values of connectivity constant and exponent on GM fractals with $2 \leq b \leq 8$

b	a_{01}	a_{02}	a_{03}	c_0	c_1	c_2	c_3
2	2	0	2	1	2	1	6
3	14	4	22	12	16	4	32
4	128	82	212	152	168	48	352
5	1532	1482	2704	2544	2120	424	4048
6	23812	31518	42368	52072	35152	6568	67680
7	486284	817798	878168	1340536	735312	112088	1374944
8	12778136	26422308	23141696	43647128	20117360	2742936	37493824
9	438476480	1079408072	797598000	1797330104	705340848	167728048	1301033984

a division, a gasket can consists of a constant number of sub triangles, the gasket itself can be constructed by combining the available triangles.

The algorithm used here maps the elementary walks to elementary triangles and combines them in a lower triangular matrix of elementary triangles, thereby mimicking the above given structure. As the elementary walks are predefined, the elementary triangles are defined accordingly. In this way, there exist six predefined triangles, each constructed from a separate elementary walk, to be used for constructing the gasket. The division of the gasket determines the dimensions of the lower triangular matrix, which is a square matrix, hence both dimensions are equal, and the total number of triangles to use in constructing the gasket.

The algorithm attempts to build a gasket by placing elementary triangles in the triangular matrix. The process starts from the top-most, left-most cell of the triangular matrix and proceeds recursively to build the gasket. Each call of the recursive function places a single triangle at the position of the call and proceeds to call the same function on the next position in the triangular matrix. The recursive calling ends when either the last cell of the matrix is reached or no suitable elementary triangle can be placed at the specified position.

Elementary triangles are arranged in such a way that the HWs extruding from previously placed elementary triangles are continued by newly placed elementary triangle rotated and/or mirrored to fit in an adequate way to satisfy the continuation requirement.

After a gasket is constructed, a final check is performed to determine whether the gasket is of the targeted type. This check is performed in such a way that three points of the triangle are checked for matches against the target triangle. If they match, the gasket is checked for loops since certain triangle combinations can result in loops. If no loops exist, the given gasket is determined to be of the targeted type and the corresponding HW counter is increased.

Optimization of the algorithm is reflected in the triangle-building approach, as certain paths are eliminated, which would otherwise be unnecessarily evaluated. The approach does introduce the path looping problem which can efficiently be dealt with by a final check of looping. This check is done in linear time and is executed by a single thread.

4.1 Parallelization and Scalability

We have implemented two approaches to parallelization of aforementioned implementation. Both approaches are based on OpenMP multi-threaded shared memory model[13].

The first approach is relatively simple but comes with some caveats as it is based on OpenMP nested parallelism. Upon calling a function on a certain cell of the matrix, a set of maximum of six elementary triangles can be placed at the given position. For each of these possible placements a new thread is spawned and the current state of the triangular matrix copied and assigned to it. This way, the number of threads increases exponentially with each call of the recursive function. Certain constructs of triangles can efficiently be resolved by a single thread, so for these constructs only a single thread is used (this refers to the non-necessity of branching for constructing the final n triangles). There are two issues with this approach. First one is the fact that just creating a large number of threads negatively impacts performance, event though not all threads are active at the same time. Second issue is that if we try to keep number of simultaneously active threads, as well as newly created threads, close to a reasonable number we pay the penalty close to the end of calculation when few long-lasting threads effectively reduce the scalability of the implementation. Careful fine-tuning allowed us to achieve very good scalability for up to 48 CPU cores but at the price of creating several thousands of threads.

While nested parallelism approach did produce scalability, it also caused severe problems with different compilers, versions and architectures available at various HP-SEE HPC resource centers. HP-SEE project represents a continuation of series of GRID and HPC related projects in SEE region [14]. In order to alleviate these problems we implemented a version that uses OpenMP 3.0 task construct. This allowed us to keep the number of threads close to the number of physical CPU cores while improving the scalability (Table 2).

Scalability testing was performed at Pecs SC resource center. Pecs supercomputer is a SGI 1000 Ultraviolet supercomputer based around Intel Xeon X7542 6-core processors with ccNUMA SMP architecture. The application was compiled by Intel C++ compiler version 12.1.5 and GCC C/C++ compiler version 4.3.4 both with -O3 optimization level. While GCC did produce measurably

Table 2. Scalability testing of SFHG application

CPU cores	Walk type	Level	CPU time/Wall time	Efficiency
1	C	8	1.00	1.00
2	C	8	1.97	0.99
4	C	8	3.91	0.98
8	C	8	7.82	0.98
12	C	8	11.31	0.94
16	C	8	15.02	0.94
24	C	8	21.74	0.91
48	C	8	40.93	0.85

better results for nested parallelism version, incomplete support for OpenMP task construct in available version prevented us from comparing the compilers for our production code.

5 Summary and Conclusion

We have analyzed asymptotic behavior of the numbers of open and closed Hamiltonian walks on Given-Mandelbrot (generalized Sierpinski gasket) fractal families and made a few new steps (8 and 9) in exact evaluating the numbers of self-avoiding walks and calculating the asymptotic behavior of leading terms. Potential capability of program in parallel processing will give the analyzing tool for more complex 3D fractal structures.

Acknowledgement. This work makes use of results produced by the High-Performance Computing Infrastructure for South East Europe's Research Communities (HP-SEE), a project co-funded by the European Commission (under contract number 261499) through the Seventh Framework Programme. HP-SEE involves and addresses specific needs of a number of new multi-disciplinary international scientific communities (computational physics, computational chemistry, life sciences, etc.) and thus stimulates the use and expansion of the emerging new regional HPC infrastructure and its services.

References

1. Vanderzande, C.: Lattice Models of Polymers. Cambridge University Press (1998)
2. Madras, N., Slade, G.: The Self Avoiding Walk. Birkäuser (1993)
3. Kondev, J., Jacobsen, J.L.: Phys. Rev. Lett. 81, 2922 (1998)
4. Duplantier, B., David, F.: J. Stat. Phys. 51, 327 (1988)
5. Owczarek, A.L., Prellberg, T., Brak, R.: Phys. Rev. Lett. 70, 951 (1993)
6. Stajić, J., Elezović-Hadžić, S.: J. Phys. A: Math. Gen. 38, 5677 (2005)
7. de Gennes, P.G.: Scaling Concepts in Polymer Physics. Cornell University Press (1979)
8. Elezović-Hadžić, S., Marčetić, D., Maletić, S.: Phys. Rev. E 76, 011107 (2007)
9. Given, J.A., Mandelbrot, B.B.: J. Phys. A: Math. Gen. 16, L565 (1983)
10. Dhar, D., Singh, Y.: Statistics of Polymers in Random Media. World Scientific, Singapore (2005)
11. Bradley, R.M.: J. Phys. A: Math. Gen. 22, L19 (1989)
12. Orland, H., Itzykson, C., de Dominicis, C.: J. Physique 46, L353 (1985)
13. OpenMP, `http://openmp.org/wp/`
14. Balaz, A., Prnjat, O., Vudragovic, D., Slavnic, V., Liabotis, I., Atanassov, E., Jakimovski, B., Savic, M.: J. Grid Comput. 9, 135 (2011)