A Fast Ergodic Algorithm for Generating Ensembles of Equilateral Random Polygons

R. Varela[‡], K. Hinson[†], J. Arsuaga[*][§], and Y. Diao[†][‖]

[‡]Department of Computer Science
San Francisco State University
1600 Holloway Ave
San Francisco, CA 94132
[†]Department of Mathematics and Statistics
University of North Carolina at Charlotte
Charlotte, NC 28223, USA
[*]Department of Mathematics
San Francisco State University
1600 Holloway Ave
San Francisco, CA 94132

Abstract. Knotted structures are commonly found in circular DNA and along the backbone of certain proteins. In order to properly estimate properties of these three dimensional structures it is often necessary to generate large ensembles of simulated closed chains (i.e. polygons) of equal edge lengths (such polygons are called equilateral random polygons). However finding efficient algorithms that properly sample the space of equilateral random polygons is a difficult problem. Currently there are no proven algorithms that generate equilateral random polygons with its theoretical distribution. In this paper we propose a method that generates equilateral random polygons in a "step-wise uniform" way. We prove that this method is ergodic in the sense that any given equilateral random polygon can be generated by this method and we show that the time needed to generate an equilateral random polygon of length n is linear in terms of n. These two properties make this algorithm a big improvement over the existing generating methods. Detailed numerical comparisons of our algorithm with other widely used algorithms are provided.

AMS classification scheme numbers: Primary 57M25, secondary 92B99.

[§] To whom correspondence regarding the computational aspect of the paper should be addressed (jarsuaga@math.sfsu.edu)
[‖] To whom correspondence regarding the theoretical aspect of the paper should be addressed (ydiao@uncc.edu)

1. Introduction

Knots and links (i.e. interlocked rings) are commonly found in nucleic acids and proteins. DNA knots and links appear as the product of random cyclization reactions of DNA molecules in solution [43, 46] and in confinement [2], as products of enzyme mediated biochemical reactions such as those mediated by site-specific recombinases [6, 8, 9, 44, 51], topoisomerases [7, 10, 22, 55] and condensins [25, 40, 49] and as nanotechnological devices [35, 45]. Furthermore they are also found in some biological systemssuch as some bacterial and plant viruses [3, 31] and bacteria harvoring mutations in their topoisomerases [47]. On the other hand, links are found in newly replicated bacterial chromosomes [21] as well as in mitochondrial DNA from trypanosomes (reviewed in [28]). Knots and links are also found along the backbone of some proteins. Recent studies of some protein crystal structures from viruses [57], bacteria [26] and humans [52] have revealed knotted structures in a wide variety of enzymes such as RNA methyltransferases [38], kinases [57] and transmembrane protein [26]. Knots have posed a new paradigm in protein folding and may have important functional and evolutionary implications [29, 30, 50, 52]. A few examples of linked protein rings have been reported and they are believed to provide stability to the complex they are part of. These include the proteins that form the capsid of certain viruses (e.g. [56]), proteins in thermofilic organisms [5] as well as some engineered proteins [4].

In order to analyze and understand these biological data it is often necessary to generate large ensembles of simulated and non-correlated circular molecules [1, 3, 27, 29, 34, 52, 54], therefore fast and reliable algorithms to generate non-correlated random polygons are needed.

The simplest representation of a circular molecule is by a closed equilateral random walk in 3-space (beginning at the origin) with n edges (length n) [19, 20], where each edge represents one or more monomers [32]. There are a few algorithms that generate ensembles of such closed equilateral random walks. These include the crankshaft algorithm [27, 36] and the hedgehog algorithm [27, 41]. In the crankshaft algorithm, one starts from the regular n-gon (with unit edge length) in the plane. At any given step, two points in the polygon are selected at random. These two points define an axis that separates the polygon into two chains. One of the chains is selected at random and rotated around the axis by a random angle (the segments are allowed to cross each other in this process). The advantage of this method is that it has been shown to be ergodic [36]. That is, any possible configuration of an equilateral random polygon can be generated by this method. However, due to the high correlations of the edges generated this way, the above rotation process must be repeated $O(n)$ times to effectively eliminate any obvious correlation. Consequently, the run time needed for this algorithm to generate an equilateral random polygon of length n is on the order of $O(n^2)$. The hedgehog algorithm (described later), on the other hand, is faster (with a run time of $O(n)$ for generating an equilateral random polygon of length n), but it is unknown whether it is ergodic.

In this paper, we propose a new method for generating equilateral random polygons that is ergodic and fast. In Section 2 we give the formal definition of equilateral random polygons and provide a brief list of known theoretical results on equilateral random polygons. These results will be used to verify our numerical results in Section 5. In Section 3 we give a detailed description of our new algorithm. We then provide a proof that this algorithm is ergodic and that the computation time increases linearly with the length of the polygon (Section 4). In Section 5 we provide numerical results obtained using our new generating algorithm and compare them with the corresponding theoretical results or results obtained using the crankshaft algorithm. While we are not able to prove that our method samples the space of equilateral polygons uniformly and the polygons generated are non-correlated,

2. Basic facts about equilateral random polygons

Let us formally define the equilateral random polygons first. Suppose Y_1, Y_2, ... , Y_n are n independent random vectors uniformly distributed on S^2 (so the joint probability density function of the three coordinates of each Y_j is simply $\frac{1}{4\pi}$ on the unit sphere and 0 otherwise). An equilateral random walk of n steps, denoted by EW_n, is defined as the sequence of points in the three dimensional space $\mathbf{R}^3$: $X_0 = O$, $X_k = Y_1 + Y_2 + \cdots + Y_k$, $k = 1, 2, ..., n$. Each X_k is called a *vertex* of the EW_n and the line segment joining X_k and X_{k+1} is called an *edge* of EW_n (which is of unit length). If the last vertex X_n of EW_n is fixed, then we have a conditioned random walk $EW_n|X_n$. In particular, EW_n becomes a polygon if $X_n = O$. In this case, it is called an equilateral random polygon and is denoted by EP_n. The joint probability density function $f(X_1, X_2, ..., X_n)$ of the vertices of an EW_n is $f(X_1, X_2, ..., X_n) = \varphi(U_1)\varphi(U_2)\cdots\varphi(U_n) = \varphi(X_1)\varphi(X_2 - X_1)\cdots\varphi(X_n - X_{n-1})$.

Let X_k be the k-th vertex of an EW_n ($n \geq k > 1$), its density function $f_k(X_k)$ is defined by the integral

$$\int \varphi(X_1)\varphi(X_2 - X_1)\cdots\varphi(X_k - X_{k-1})dX_1 dX_2 \cdots dX_{k-1} \tag{1}$$

and has the closed form $f_k(X_k) = \frac{1}{2\pi^2 r}\int_0^\infty x \sin rx \left(\frac{\sin x}{x}\right)^k dx$ [42]. In the case of EP_n, the density function of the vertex X_k can be approximated by a Gaussian distribution, as given in the following theorem.

Theorem 1 *[13, 16, 17] Let X_k be the k-th vertex of an EP_n and let h_k be its density function, then*

$$h_k(X_k) \approx \left(\sqrt{\frac{3}{2\pi\sigma_{nk}^2}}\right)^3 \exp\left(-\frac{3|X_k|^2}{2\sigma_{nk}^2}\right), \tag{2}$$

where $\sigma_{nk}^2 = \frac{k(n-k)}{n}$ and the error of the estimation is at most of the order of $O\left(\frac{1}{k^{5/2}} + \frac{1}{(n-k)^{5/2}}\right)$.

This tells us that the distribution of X_k of an EP_n is approximately Gaussian. From this theorem one can then derive some important results concerning equilateral random polygons which can be used to check how likely a generating algorithm is producing equilateral random polygons with the correct distributions. One such result is listed in the following corollary.

Corollary 1 *Assume that $n = 2k$ is even and let $r = |X_k|$, which is the distance between the origin and the middle vertex X_k of the polygon. Then r has an estimated probability density function*

$$g'(r) \approx 4\pi r^2 \left(\sqrt{\frac{6}{\pi n}}\right)^3 \exp\left(-\frac{6r^2}{n}\right). \tag{3}$$

In particular, the mean of r is approximately $\sqrt{2n/3\pi}$.

From Theorem 1 it is also fairly easy to see that the mean square radius of gyration for EP_n is on the order $O(n)$. Another quantity we will use for checking the validity of our algorithm is the mean ACN of EP_n, which is defined as the following. If we project an EP_n onto a plane along a given direction, we can count the number of crossings that are visible in this particular projection. To be independent of the choice of a particular projection, we average these crossing numbers over all projections and the number so obtained is called the *average crossing number* (ACN). In fact, ACN is an important quantity since it is a natural geometric measure of polymer entanglement as it refers to the actual number of crossings that can be perceived while observing a non-perturbed trajectory of a given polymer or DNA [24]. The following result reveals the asymptotic behavior of the mean ACN. Almost perfect matching numerical results were given in the same paper containing this result.

Theorem 2 *[14] Let χ_n be the ACN of an equilateral random walk of n steps; then*

$$E(\chi_n) = \frac{3}{16} n \ln n + O(n).$$

On the other hand, Theorem 1 also enables us to obtain theoretical results regarding the topological aspects of EP_n such as the following theorem, which have been confirmed by many independently carried out simulations.

Theorem 3 *[13] Let $\mathcal{K}$ be any knot type, then there exists a positive constant ϵ such that EP_n contains $\mathcal{K}$ as a connected sum component with a probability at least $1 - \exp(-n^\epsilon)$, provided that n is large enough.*

These theoretical results, as well as those well-documented numerical results obtained using the existing generating methods, provide a solid background for us to examine the validity of our new method.

3. The generalized hedgehog method

In this section we give a detailed description of the generalized hedgehog method, a new algorithm for generating the equilateral random polygons.

3.1. Single and double rotation operations

Two operations are needed in the generalized hedgehog method. The first one, called a *single rotation*, is used in the original hedgehog method. We provide it here for the convenience of our reader.

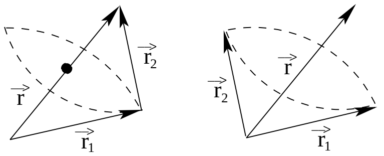

Figure 1. A single rotation involving two unit vectors.

Definition 1 *Given two unit vectors $\vec{r}_1$ and $\vec{r}_2$, a single rotation involving $\vec{r}_1$ and $\vec{r}_2$ is a rotation of $\vec{r}_1$ and $\vec{r}_2$ around the axis $\vec{r} = \vec{r}_1 + \vec{r}_2$ such that the rotation angle is uniformly chosen between* 0 *and* 2π*. This rotation can be viewed in two ways as shown in Figure 1.*

Let $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ be three unit vectors (here we consider all vectors as rooted at the original point O). Let $\vec{r}_1 = \overrightarrow{OX}$, $\vec{r}_1 + \vec{r}_2 = \overrightarrow{OY}$ and $\vec{r}_1 + \vec{r}_2 + \vec{r}_3 = \overrightarrow{OZ}$. In the following we describe a procedure that replaces $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ with three new unit vectors $\vec{r'}_1$, $\vec{r'}_2$ and $\vec{r'}_3$ such that $\vec{r}_1 + \vec{r}_2 + \vec{r}_3 = \vec{r'}_1 + \vec{r'}_2 + \vec{r'}_3$. Let $\vec{r'}_1$, $\vec{r'}_2$ and $\vec{r'}_3$ be three unit vectors such that $\vec{r'}_1 + \vec{r'}_2 + \vec{r'}_3 = \vec{r}$ (keep in mind that $\vec{r} = \overrightarrow{OZ} = \vec{r}_1 + \vec{r}_2 + \vec{r}_3$ is a fixed vector in this process). The end points of the vectors $\vec{r'}_1$, $\vec{r'}_1 + \vec{r'}_2$ and $\vec{r'}_1 + \vec{r'}_2 + \vec{r'}_3 (= \vec{r})$ define a 4-sided polygon with side lengths 1, 1, 1 and $r = |\vec{r}| = |\overrightarrow{OZ}|$. In the case that $\vec{r'}_1 = \vec{r'}_2$, we obtain a triangle of side lengths 1, 2 and r as shown in Figure 2.

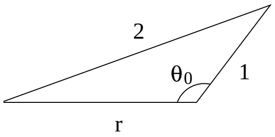

Figure 2. The triangle that defines the maximum angle θ_0 for θ.

In this case, the angle θ_0 between the vector $\vec{r}$ and $\vec{r'}_3$ is given by the following formula

$$\theta_0 = \begin{cases} \cos^{-1}\left(\frac{r^2-3}{2r}\right), & r > 1 \\ \pi, & r \leq 1, \end{cases}$$

Notice that in general, the angle θ between $\vec{r}$ and $\vec{r'}_3$ is less than or equal to θ_0. We will now choose a point Y' uniformly on the spherical region defined by $\{U \in S(Z) : \theta \leq \theta_0\}$, where $S(Z)$ is the unit sphere centered at Z and θ is the smaller angle between $\overrightarrow{ZO}$ and $\overrightarrow{ZU}$. Let $\vec{r'}_3 = \overrightarrow{OZ} - \overrightarrow{OY'}$. Now $S(O)$ and $S(Y')$ intersect in a circle and we will choose a point X' on it uniformly. See Figure 3 for an illustration of this process.

Finally, we define $\vec{r'}_1 = \overrightarrow{OX'}$ and $\vec{r'}_2 = \overrightarrow{OY'} - \overrightarrow{OX'}$. Thus we have replaced the three unit vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ with $\vec{r'}_1$, $\vec{r'}_2$ and $\vec{r'}_3$. We call this operation a *double rotation* of the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$. Let us emphasize that a double rotation involving $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$ does not change their sum $\vec{r}_1 + \vec{r}_2 + \vec{r}_3$ since $\vec{r}_1 + \vec{r'}_1 + \vec{r'}_3 = \overrightarrow{OX'} + (\overrightarrow{OY'} - \overrightarrow{OX'}) + (\overrightarrow{OZ} - \overrightarrow{OY'}) = \overrightarrow{OZ}$.

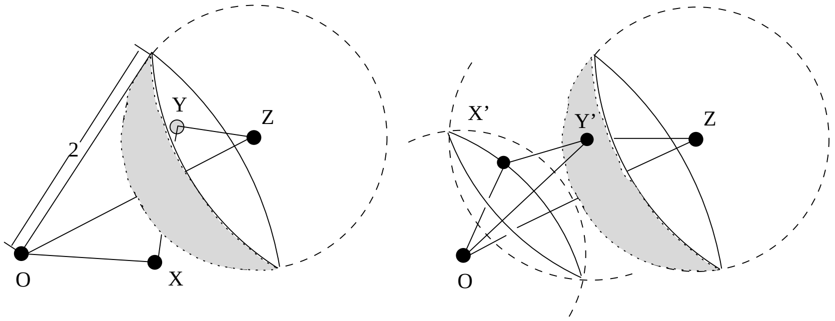

Figure 3. The determination of Y' and X'.

3.2. The algorithm

We will now describe the algorithm to be used to generate a random equilateral polygon of length n. Depending on n is odd or even, the first step is slightly different.

Step 1_1. If n is odd, then we will generate a random unit vector $\vec{r}_1$ uniformly over the unit sphere starting from the origin O. Let the end point of $\vec{r}_1$ be X. We then generate a random unit vector $\vec{r}_2$ starting from X by choosing its ending point Y uniformly on the unit circle which is the intersection of the two unit sphere $S(O)$ and $S(X)$. Finally we let $\vec{r}_3 = \overrightarrow{YO}$. This gives us a random equilateral polygon of length 3 since $\vec{r}_1 + \vec{r}_2 + \vec{r}_3 = \vec{0}$.

Step 1_2. If n is even, then we first randomly choose two unit vectors $\vec{r}_1$, $\vec{r}_2$ uniformly on the unit sphere $S(O)$ and perform a single rotation involving $\vec{r}_1$ and $\vec{r}_2$. Combining the two resulting vectors with $\vec{r}_3 = -\vec{r}_1$ and $\vec{r}_4 = -\vec{r}_2$ gives us a random equilateral polygon of length 4. Let us name the resulting vectors still with the names $\vec{r}_1$ to $\vec{r}_4$.

Step 2. Randomly generate a unit vector $\vec{r}$ (uniformly on the unit sphere). Randomly choose two vectors $\vec{r'}$ and $\vec{r''}$ from the previous list and replace them with $-\vec{r}$ and the three vectors resulted from a double rotation on $\vec{r}$, $\vec{r'}$ and $\vec{r''}$. By the nature of the double rotation, the sum of all the vectors in the list is always the zero vector at any step. This step adds two vectors and the result is an equilateral polygon of 4 or 5 edges.

Step 2 may now be repeated and at step k we arrive at a list of either $n = 2k + 1$ or $n = 2k + 2$ unit vectors, depending on what we did at step 1. The list is then randomly shuffled. By a misuse of notation, let us again name the vectors in the final list as $\vec{r}_1$, $\vec{r}_2$, ..., $\vec{r}_n$. Now the line segments joining the end points of $\vec{r}_1$, $\vec{r}_1 + \vec{r}_2$, ..., $\vec{r}_1 + \vec{r}_2 + \cdots + \vec{r}_n = \vec{O}$ define an equilateral polygon of n edges.

4. The ergodicity of the generalized hedgehog algorithm

In this section, we show that the generalized hedgehog method introduced in the last section is ergodic. That is, for any given equilateral polygon P_n of length n, there exist a set of n unit vectors $\vec{r}_1$, $\vec{r}_2$, ..., $\vec{r}_n$ such that P_n can be obtained by performing the algorithm described in the last section on these vectors. First, we need the following lemma.

Lemma 1 *Let $\vec{r}_1$, $\vec{r}_2$, ..., $\vec{r}_n$ be any $n \geq 4$ unit vectors such that* $\vec{r}_1 + \vec{r}_2 + \cdots + \vec{r}_n = \vec{0}$*, then there exist four distinct positive integers k_1, k_2, k_3, k_4 between 1 and n such that $|\vec{r}_{k_1} + \vec{r}_{k_2} + \vec{r}_{k_3} + \vec{r}_{k_4}| < 2$.*

Proof. Assume the contrary, then for any distinct integers k_1, k_2, k_3, k_4 between 1 and n, we have $|\vec{r}_{k_1} + \vec{r}_{k_2} + \vec{r}_{k_3} + \vec{r}_{k_4}| \geq 2$. Let $\vec{r}_j = (x_j, y_j, z_j)$. By the given condition, we have

$$x_j^2 + y_j^2 + z_j^2 = 1$$

for each $1 \leq j \leq n$ and

$$\sum_{j=1}^{n} x_j = 0, \ \sum_{j=1}^{n} y_j = 0, \ \sum_{j=1}^{n} z_j = 0.$$

Square both sides of the above equations and sum over the results, we obtain

$$n + \sum_{i \neq j} (x_i x_j + y_i y_j + z_i z_j) = 0.$$

Therefore,

$$\sum_{i \neq j} (x_i x_j + y_i y_j + z_i z_j) = -n < 0. \tag{4}$$

On the other hand, since $|\vec{r}_{k_1} + \vec{r}_{k_2} + \vec{r}_{k_3} + \vec{r}_{k_4}| \geq 2$, we have

$$\begin{aligned} & (x_{k_1} + x_{k_2} + x_{k_3} + x_{k_4})^2 \\ + & (y_{k_1} + x_{k_2} + y_{k_3} + y_{k_4})^2 \\ + & (z_{k_1} + z_{k_2} + z_{k_3} + z_{k_4})^2 \geq 4. \end{aligned}$$

This implies that

$$\begin{aligned} & (x_{k_1}x_{k_2} + x_{k_1}x_{k_3} + x_{k_1}x_{k_4} + x_{k_2}x_{k_3} + x_{k_2}x_{k_4} + x_{k_3}x_{k_4}) \\ + & (y_{k_1}y_{k_2} + y_{k_1}y_{k_3} + y_{k_1}y_{k_4} + y_{k_2}y_{k_3} + y_{k_2}y_{k_4} + y_{k_3}y_{k_4}) \\ + & (z_{k_1}z_{k_2} + z_{k_1}z_{k_3} + z_{k_1}z_{k_4} + z_{k_2}z_{k_3} + z_{k_2}z_{k_4} + z_{k_3}z_{k_4}) \geq 0. \end{aligned}$$

Sum both sides of the above inequality for all possible k_1, k_2, k_3 and k_4, we obtain

$$\sum_{i \neq j} (x_i x_j + y_i y_j + z_i z_j) \geq 0,$$

which contradicts (4).

We are now ready to prove the following ergodicity theorem.

Theorem 4 *The generalized hedgehog method is ergodic, that is, any configuration of an equilateral polygon of n edges can be constructed from this algorithm.*

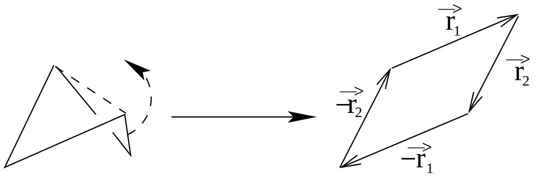

Figure 4. The case of a polygon with 4 edges

Proof. For the simple case of $n = 3$, there is nothing to prove since any random equilateral random polygon of length can be uniformly selected at Step 1 in our algorithm. If P is an equilateral polygon of length 4, we can perform a single rotation to put all four vectors in the same plane as shown in Figure 4. Apparently, these are two pairs of opposite vectors $\vec{r}_1$, $-\vec{r}_1$, $\vec{r}_2$ and $-\vec{r}_2$. In other words, P can be obtained by choosing $\vec{r}_1$ and $\vec{r}_2$ first, followed by a single rotation involving $\vec{r}_1$ and $\vec{r}_2$ and finally combining the resulting vectors with $\vec{r}_3 = -\vec{r}_1$ and $\vec{r}_4 = -\vec{r}_2$.

Let us now assume that we have proven for any equilateral polygon P_k of length k such that $3 \leq k \leq n-1$, there exists k unit vectors $\vec{r}_1$, $\vec{r}_2$, ..., $\vec{r}_k$ such that P_k can be obtained by the generalized hedgehog algorithm involving these vectors, where $n-1 \geq 4$. We will now consider an equilateral polygon P_n of length n. Notice that P_n is defined by n consecutive unit vectors $\vec{r}_1$, $\vec{r}_2$, ..., $\vec{r}_n$ such that $\vec{r}_1 + \vec{r}_2 + \cdots + \vec{r}_n = \vec{0}$. By Lemma 1, there exist four distinct integers k_1, k_2, k_3 and k_4 between 1 and n such that $|\vec{r}_{k_1} + \vec{r}_{k_2} + \vec{r}_{k_3} + \vec{r}_{k_4}| < 2$. Without loss of generality (since we are allowed to shuffle the order of the vectors in the algorithm), let us assume that $k_1 = 1$, $k_2 = 2$, $k_3 = 3$ and $k_4 = 4$. That is, $|\vec{r}_1 + \vec{r}_2 + \vec{r}_3 + \vec{r}_4| < 2$. Let U be the end point of $\vec{r}_1 + \vec{r}_2 + \vec{r}_3 + \vec{r}_4$, Z be the end point of $\vec{r}_1 + \vec{r}_2 + \vec{r}_3$, Y be the end point of $\vec{r}_1 + \vec{r}_2$ and X be the end point of $\vec{r}_1$, as shown in Figure 5.

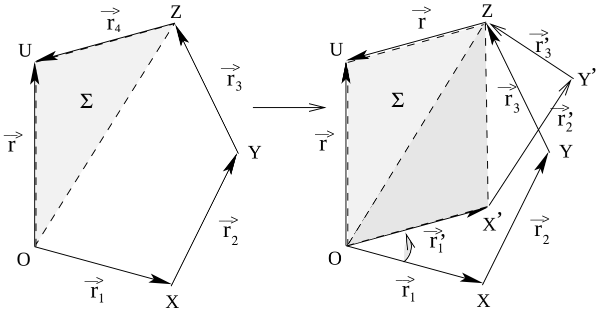

Figure 5. Creating the opposite pair

We will now perform a double rotation such that the resulting $\vec{r'}_1 = -\vec{r}_4$. We see that this can be done since $\overline{OU} = \overline{X'Z} < 2$, as shown in Figure 5. Let us name the resulting vectors $\vec{r'}_1$, $\vec{r'}_2$ and $\vec{r'}_3$ respectively. Apparently, eliminating $\vec{r'}_1$ and $\vec{r}_4$ will give us an equilateral polygon P_{n-2} of length $n-2$. In other words, P_n can be obtained by selecting $\vec{r'}_1$ first, then performing a double rotation involving $\vec{r'}_1$, $\vec{r'}_2$ and $\vec{r'}_3$ (the latter two are edges of P_{n-2}), followed by adding in the vector $\vec{r}_4 = -\vec{r'}_1$. By our assumption, P_{n-2} can be obtained by the algorithm. Thus P_n can also be obtained by the algorithm.

5. Run time consideration and numerical comparisons with existing methods

In order to validate our algorithm we first tested the computational time that needed to generate a sample of 50,000 polygons. Next we analyzed distributions of geometrical and topological properties commonly associated to closed circular molecules. These include the three dimensional distribution the of points on the polygon, the radius of gyration, the knotting probability and the mean average crossing number. In all experiments described below 100,000 polygons were generated for each polygon size and sizes ranged from 20 to 1000 in increments of 20.The quantities described were calculated for each molecule and averaged over all the samples.

- **Run time determination and comparison**

First we confirmed that the time complexity of the algorithm is linear with the length of the polygon (see Section 3). The following figure summarizes our result. It shows that the time needed to generate an equilateral random polygon of length n following $y = 0.31n-1.030$. By comparison, the time needed to generate an equilateral random polygon of length n using the crankshaft method shows a growth rate of at least $O(n^2)$.

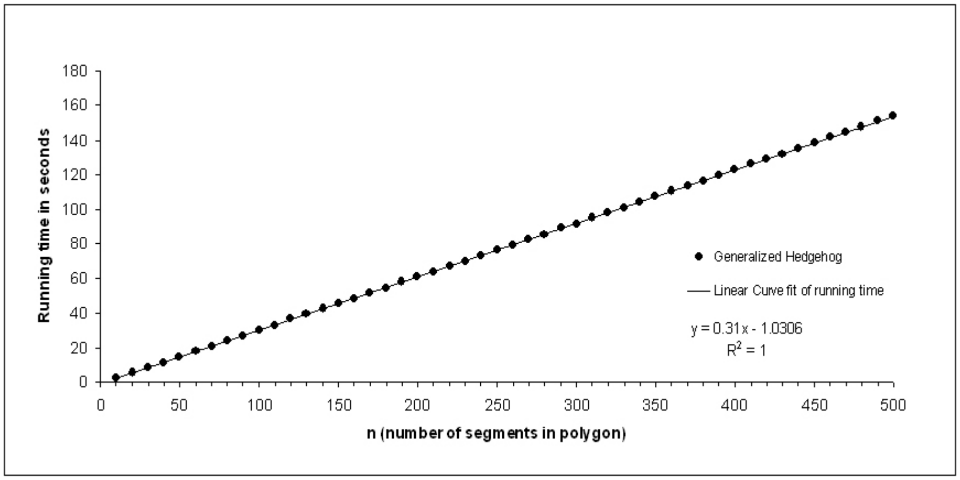

Figure 6. Linear running time: The x-axis represents the length of the polygon and the y-axis the time (in seconds) needed by the algorithm to generate 50,000 polygons.

- **Distributions**

First we investigated the average spatial distribution occupied by the points in the polygons generated by our algorithm. We computed the average distance from the k-th vertex (of an equilateral random polygon of length n) to the origin (the 0-th vertex) with varying k values. In this calculation we expect that the average distance from the middle vertex $X_{n/2}$ of an equilateral random polygon of length n to the origin to be the maximum among the k values. Furthermore, according to Corollary 1, this average distance is estimated by $\approx \sqrt{2n/3\pi}$. Figure 7 below shows the distribution of distances for different points along the polygon. The solid dots in the figure are from the plot of the function $\approx \sqrt{2n/3\pi}$. Since the peak of each curve represents the average distance from the middle vertex to the origin, a good fit means that the dots should be at or near the peaks. This is clearly seen from the figure. Such a nice fit certainly suggests that the random polygons generated using our method are following the theoretical distribution in terms of the vertex to vertex distances. It also shows that $\approx \sqrt{2n/3\pi}$ is a very good estimator of the average distance from the middle vertex to the origin.

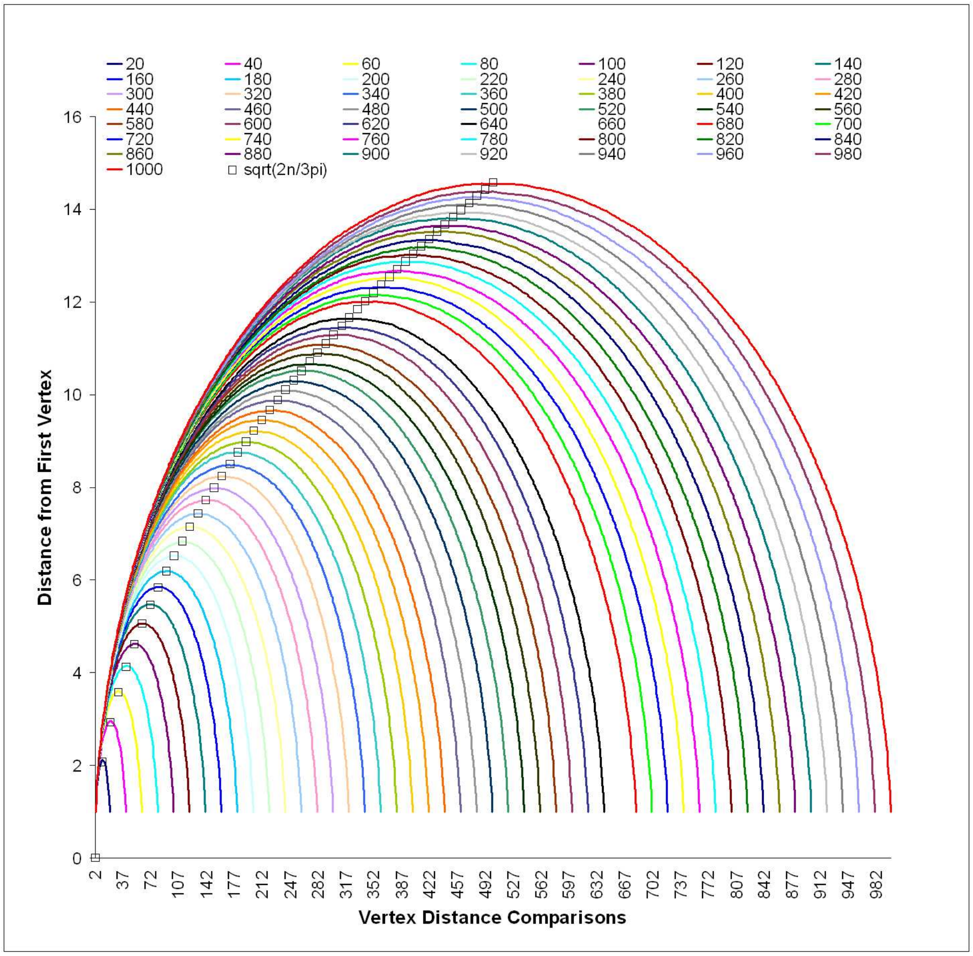

Figure 7. Average vertex distance distribution from first vertex. The x-axis represents the vertex with in the polygon from which we calculated the distance to any other vertex. The y-axis represents the average distance calculated. A χ^2 goodness of fit test with $df = 49$ yields a χ^2 value of 0.0504, indicating a near perfect fit.

In an independent numerical study, we generated 10,000 equilateral random polygons of length 200, 400, 1000 and 2000 each and computed the distance between the origin and their middle point vertex (namely X_{100}, X_{200}, X_{500} and X_{1000} respectively). We then carried out a kernel estimation for the probability density function of this distance. Recall from Corollary 1 that this probability density function can be estimated by $g'(r) \approx 4\pi r^2 \left(\sqrt{\frac{6}{\pi n}}\right)^3 \exp\left(-\frac{6r^2}{n}\right)$. Figure 8 shows the plots of the numerical pdfs obtained by the kernel estimation as well as the plots of the theoretical curve $4\pi r^2 \left(\sqrt{\frac{6}{\pi n}}\right)^3 \exp\left(-\frac{6r^2}{n}\right)$ for the cases of $n = 200$, 400, 1000 and 2000. Again, the nice fits in these numerical studies strongly suggest that the random polygons we generate are following their theoretical distributions, and that the approximating pdf function $g'(r)$ is quite accurate. It is worthwhile for us to point out that numerical studies on the approximating pdf $g'(r)$ have not been carried out before.

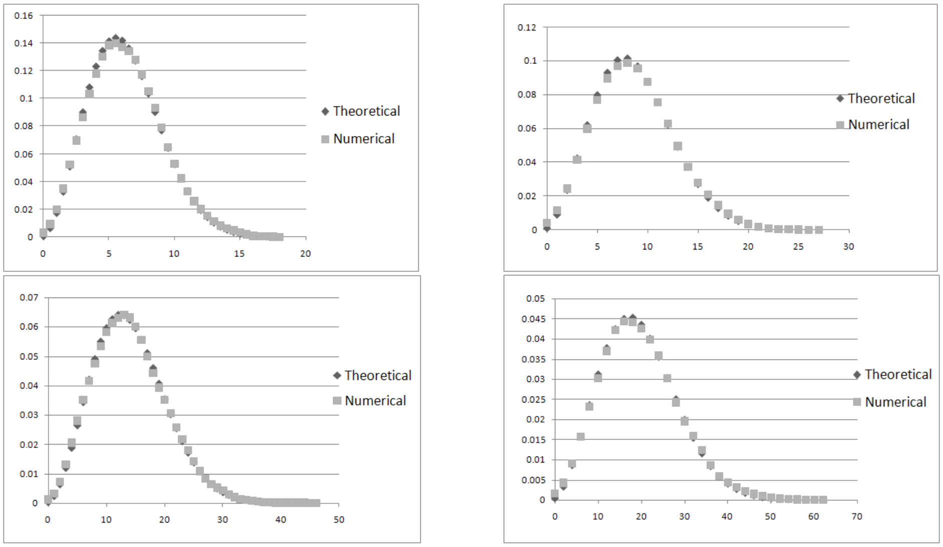

Figure 8. Kernel estimations of the pdf for the middle point distance for $n = 200$, 400, 1000 and 2000. The Gaussian kernel is used for the plots. The bandwidth for $n = 200$ is 0.5, for $n = 400$ and 1000 is 1 and for $n = 2000$ is 2.

- **Radius of gyration**

Next we studied the mean radius of gyration. The radius of gyration estimates the size of a molecule and can be experimentally measured by standard sedimentation assays. In [18] it was found that the squared radius of gyration of a polygonal molecule increases as $\frac{1}{12}n^{2\nu}$ with $\nu = 0.5$. Our result is in excellent agreement with that presented in [18] as shown in Figure 9. For comparison purposes we also included results obtained through our own implementation of the crankshaft algorithm.

- **Knotting probability**

Next we asked what is the probability that a given polygon is knotted. As stated in Theorem 3, $P(knotted) \geq 1 - \exp(-n^{\epsilon})$ for some positive constant ϵ [13]. In fact, as shown in many numerical studies long EPs tend to be knotted with a knotting probability of the form $1 - \exp(-\alpha n)$ with $\alpha \approx \frac{1}{244}$ [11, 37]. We compared our results

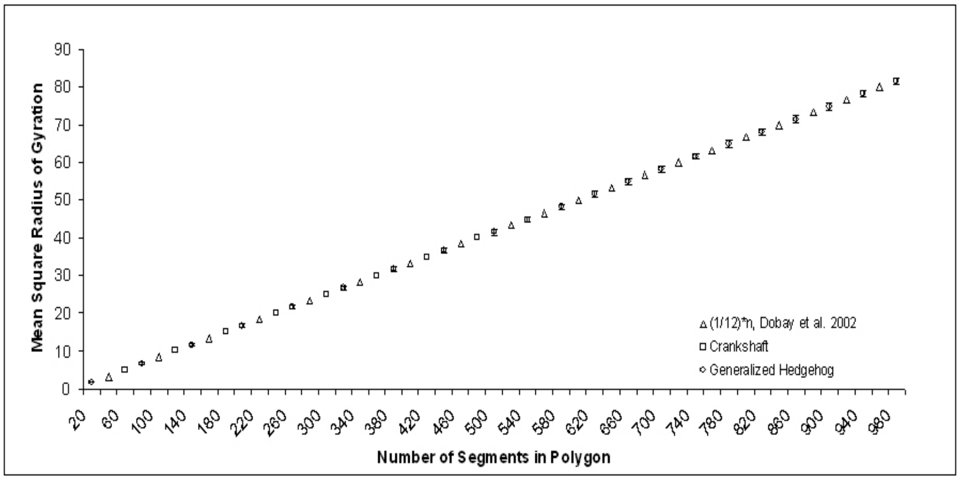

Figure 9. Mean squared radius of gyration of polygons generated using the generalized Hedgehog method with one standard deviation. The x-axis represents the number of segments within the polygon generated and the y-axis represents the mean squared radius of gyration calculated. The results by the crankshaft method and by Dobay et al. [18] are provided for comparison. The standard deviations are less than or about the size of the dots.

to those presented in Micheletti et al. [34] (where the crankshaft algorithm was used) and those calculated using our own crankshaft algorithm implementation. Figure 10 below illustrates the comparison of our experiments and shows an excellent agreement of the three methods.

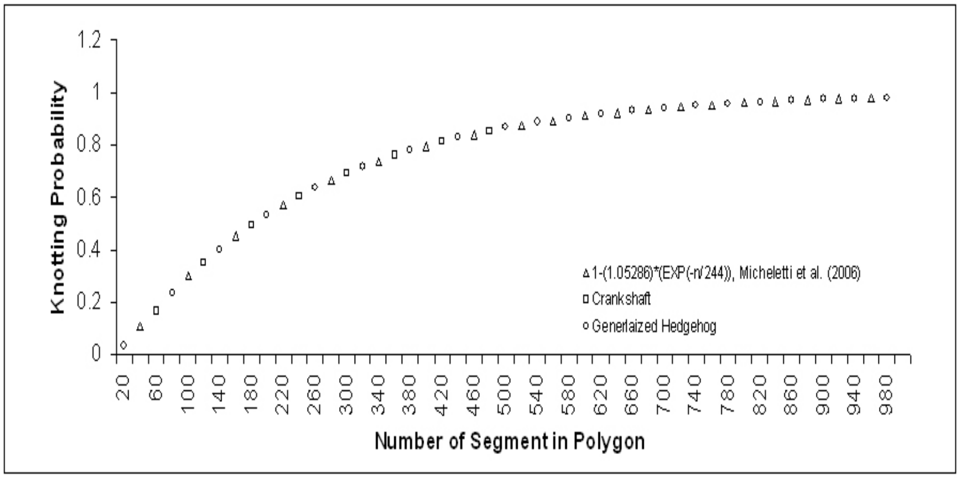

Figure 10. Knotting probability comparison. The x-axis represents the number of segments of the polygon generated and the y-axis represents the average knotting probability.

- **Average crossing number**

The average crossing number (ACN) is a geometrical measure of the entanglement complexity of the polygon and it has been proposed that knotted DNA molecules migrate in gel electrophoresis proportionally to the average crossing number of their ideal configuration [53]. The ACN is known to grow as $\frac{3}{16}n\ln(n) + O(n)$ as we mentioned earlier in Theorem 2 [14]. We compared our mean ACN values with the values obtained using our own implementation of the crankshaft method and the theoretical expected results derived from Diao et al. [14]. The results are illustrated in Figure 11.

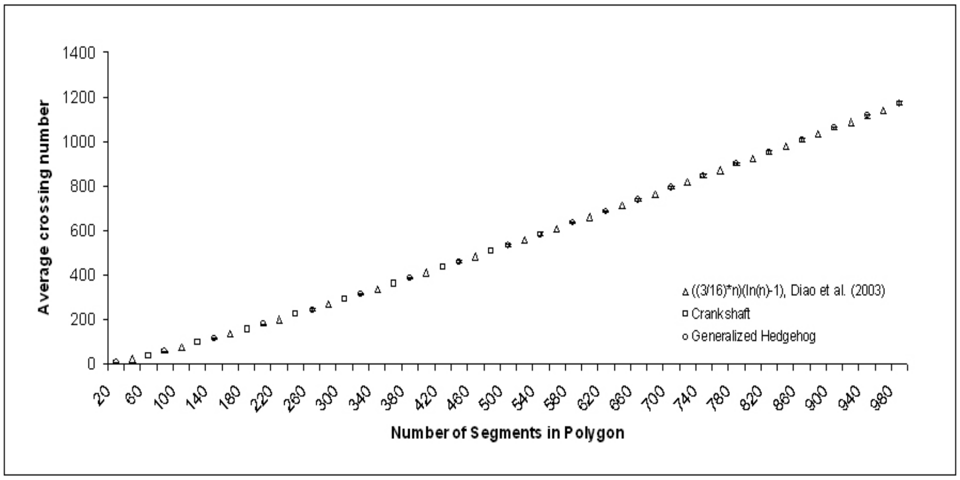

Figure 11. Mean average crossing number comparison. x-axis represents the number of segments within the polygons generated and y-axis represents the mean average crossing number calculated. The standard deviations are less than or about the size of the dots.

6. Conclusions and ending remarks

Knots and links are commonly found in nucleic acids and proteins. DNA knots and links have been used for many years in experimental laboratories and have been key to unveil the action of some DNA binding proteins (e.g. [6, 7, 25]) and the chromosome organization in certain viruses [3]. Protein knots on the other hand are novel structures [50] and hold a great promise for better understanding the problem of protein folding [30]. Importantly their true functional and evolutionary significance remains to be determined [29, 52]. In many of these structural studies it is often necessary to generate large samples of non-correlated polygonal curves. However this task has been hindered by the lack of efficient and ergodic algorithms that generate such samples. Here we have presented a new algorithm to generate large samples of independent knotted equilateral polygons. We have rigorously shown that this algorithm is ergodic and also that it can reproduce current known numerical results for the mean square radius of gyration, the knotting probability, and the mean average crossing number. Furthermore the algorithm seems to generate random equilateral random polygons according to its theoretical distribution as indicated by the tests we carried out in our numerical study. However it remains a challenging problem to prove or disprove this theoretically.

This algorithm can be used to study knotting and linking of DNA molecules in free solution. However more accurate representation of the DNA chain and proteins is needed. In future studies we will address this problems and also extend the algorithm to study problems of DNA knotting and linking in confinement.

Acknowledgement. J. Arsuaga is partially supported by NIH grant NIH-2S06GM52588-12 and Y. Diao is partially supported by NSF grant DMS-0712958.

References

[1] Arsuaga J et al 2007, *J Physics A* **40** 11697–711.
[2] ——, Vazquez M, Trigueros S, Sumners DW and Roca J 2002, *Proc Natl Acad Sci USA* **99** 5373–7.
[3] ——, Vazquez M, McGuirk P, Trigueros S, Sumners DW and Roca J 2005, *Proc Natl Acad Sci USA* **102** 9165–9.
[4] Blankenship JW and Dawson PE 2003, *J Mol Biol* **327**(2) 537–48.
[5] Boutz DR, Cascio D, Whitelegge J, Perry LJ and Yeates TO 2007, *J Mol Biol* **368**(5) 1332–44.
[6] Buck D and Flapan E 2007, *J Mol Biol* **374**(5) 1186-99.
[7] Buck GR and Zechiedrich EL 2004, *J Mol Biol* **340**(5) 933–9.
[8] Darcy IK et al 2006, *BMC Bioinformatics* **7** 435.
[9] ——and Scharein RG 2006, *Bioinformatics* **22**(14) 1790–1.
[10] Dean FB, Stasiak A, Koller T and Cozzarelli NR 1985, *J Biol Chem* **260** 4975–83.
[11] Deguchi T and Tsurusaki K 1997, *Phys Rev E* **55** 6245–8.
[12] Diao Y 1990, Ph. D. Thesis, Florida State University.
[13] ——1995, *J Knot Theory Ramifications* **4**(2) 189–96.
[14] ——, Dobay A, Kusner R, Millett K and Stasiak A 2003, *J Physics A* **36**(46) (2003), 11561–74.
[15] ——, Dobay A and Stasiak A 2005, *J Physics A* **38**(35) 7601–16.
[16] ——, Nardo J and Sun Y 2001, *J Knot Theory Ramifications* **10**(4) 597–607.
[17] ——, Pippenger N and Sumners DW 1994, *J Knot Theory Ramifications* **3**(3) 419–29.
[18] Dobay A, Dubochet J, Millett K, Sottas P-E and Stasiak A 2003, *Proc Natl Acad Sci USA* **100**(10) 5611-5.
[19] Edwards SF 1967, *Proc Phys Soc* **91** 513–9.
[20] ——1968, *J Physics A* **1** 15–28.
[21] Grainge I, Bregu M, Vazquez M, Sivanathan V, Ip SC and Sherratt DJ 2007, *EMBO J* **26**(19) 4228–38.
[22] Hsieh T 1983, *J Biol Chem* **258** 8413–20.
[23] Katritch V, Olson WK, Vologodskii AV, Dubochet J and Stasiak A 2000, *Phys Rev E* **61** 5545–9.
[24] ——, Bednar J, Michoud D, Scharein RG, Dubochet J and Stasiak A 1996, *Nature* **384** 142–5.
[25] Kimura K, Rybenkov VV, Crisona NJ, Hirano T and Cozzarelli NR 1999, *Cell* **98**(2) 239–48.
[26] King NP, Yeates EO and Yeates TO 2007,*J Mol Biol* **373**(1) 153–66.
[27] Klenin KV, Vologodskii AV, Anshelevich VV, Dykhne AM and Frank-Kamenetskii MD 1988, *J Biomolec Str and Dyn* **5** 1173–85.
[28] Liu B, Liu Y, Motyka SA, Agbo EE and Englu PT 2005, *Trends Parasitol* **21** 363–9.
[29] Lua RC and Grosberg AY 2006,*PLoS Comput Biol* **2**(5) e45.
[30] Mallam AL, Onuoha SC, Grossmann JC and Jackson SE 2008, *Mol Cell* **30** 642–8.
[31] Menissier J, de Murcia G, Lebeurier G and Hirth L 1983, *EMBO J* **2** 1067–71.
[32] Michels JPJ and Wiegel FW 1986, *Proc R Soc Lond A* **403** 269–84.
[33] ——and Wiegel FW 1982, *Phys Letts A* **90** 381–4.
[34] Micheletti C, Marenduzzo D, Orlandini E and Sumners DW 2006, *J Chem Phys* **124** 064903.1 –10.
[35] Mueller JE, Du SM and Seeman NC 1991, *J Am Chem Soc* **113** 6306–8.
[36] Millett K 2000, Knots in Hellas'98 (Delphi), Ser. Knots Everything **24** (World Scientific) 306–34.
[37] ——and Rawdon E 2005, Physical and Numerical Models in Knot Theory, Ser. Knots Everything **36** 247–74.
[38] Nureki O et al 2002,*Acta Crystallogr D Biol Crystallogr* **58**(7) 1129–37.

[39] Orlandini E, Janse van Rensburg EJ, Tesi MC and Whittington SG 1994, *J Physics A* **27**(2) 335–45.
[40] Petrushenko ZM, Lai CH, Rai R and Rybenkov VV 2006, *J Biol Chem* **281**(8) 4606–15.
[41] Plunkett P et al 2007, *Macromolecules* **40** 3860–7.
[42] Rayleigh L 1919, *Phil Mag S 6* **37**(220) 321–47.
[43] Rybenkov VV, Cozzarelli NR and Vologodskii AV 1993, *Proc Natl Acad Sci USA* **90** 5307–11.
[44] Saka Y and Vazquez M 2002, *Bioinformatics* **18** 1011–2.
[45] Seeman NC 2003, *Biochemistry* **42** 7259–69.
[46] Shaw SY and Wang JC 1993, *Science* **260** 533–6.
[47] Shishido K, Komiyama N and Ikawa S 1987, *J Mol Biol* **195**(1) 215–8.
[48] Shimamura M and Deguchi T 2002, *Phys Rev E* **66**(1) 040801.1–4.
[49] Stray JE et al 2005, *J Biol Chem* **280**(41) 34723–34.
[50] Taylor WR 2000, *Nature* **406** 916–9.
[51] Vazquez M, Colloms S and Sumners DW 2005, *J Mol Biol* **346**(2) 493–504.
[52] Virnau P, Mirny LA and Kardar M 2006, *PLoS Comput Biol* **2**(9) e122.
[53] Vologodskii AV et al 1998,, *J Mol Biol* **278**(1) 1–3.
[54] Wallin S, Zeldovich KB and Shakhnovich EI 2007, *J Mol Biol* **368**(3) 884–93.
[55] Wasserman SA and Cozzarelli NR 1991, *J Biol Chem* **266** 20567–73.
[56] Wikoff WR et al 2000, *Science* **289** 2129–33.
[57] Yeates TO, Norcross TS and King NP 2007, *Curr Opin Chem Biol* **11**(6) 595–603.