Supporting Information: Sequence-dependent persistence lengths of DNA (Mitchell, Glowacki, Grandchamp, Manning, and Maddocks)

J. of Chemical Theory and Computation (2016)
DOI: 10.1021/acs.jctc.6b00904

S.1 Computation of coarse-grain approximations to the tangent

The unit tangents $\mathbf{t}_i^{[k]}$ (where for simplicity we only take k odd) to the straight lines that are the best least squares linear approximation to a consecutive run of $(k+1)$ base-pair locations $\mathbf{r}_i, \ldots, \mathbf{r}_{i+k}$ can be computed for any configuration, $\mathbf{t}_i^{[k]}$ as follows. First calculate the (geometrical) centre of mass $\mathbf{c}_i^k = (\sum_{j=i}^{j=i+k} \mathbf{r}_j)/(k+1)$. Then $\mathbf{t}_i^{[k]}$ is the unit eigenvector (with positive projection on the chord $\mathbf{r}_{i+k} - \mathbf{r}_i$) corresponding to the largest eigenvalue of the (local gyration) matrix $\sum_{j=i}^{j=i+k} (\mathbf{r}_j - \mathbf{c}_i^k) \otimes (\mathbf{r}_j - \mathbf{c}_i^k)$. The case $k = 1$ reduces analytically to the unit tangent to the junction chord between two consecutive base pair origins $\mathbf{t}_i^{[1]} = (\mathbf{r}_{i+1} - \mathbf{r}_i)/\left\|\mathbf{r}_{i+1} - \mathbf{r}_i\right\|$, while nonlocal coarse grain choices $k > 1$ must be computed numerically.

S.2 Details regarding the cgDNAmc code

S.2.1 Downloading the software

The C++ code *cgDNAmc*, along with two libraries it depends upon, *algebra3d* and *cgDNArecon*, is freely available with online instructions on how to download, compile, and run it.[1] The user has to supply any desired problem-specific, post-processing code fragments implementing specialised techniques such as the sliding-window average used in modelling cryo-EM experimental data.

The remainder of this section describes our Monte Carlo implementation in further detail. The simulations described here are not particularly intensive, nevertheless we have taken some efforts to make *cgDNAmc* code efficient. Benchmark results presented below were obtained on a mid-range laptop computer.

S.2.2 Direct Monte Carlo sampling

As described in the main text, a key step in our direct Monte Carlo sampling is the Cholesky decomposition $\mathbf{K} = \mathbf{L}\mathbf{L}^T$ of the sparse stiffness matrix $\mathbf{K}$. For efficient sampling, the key property of the Cholesky factorization is that if $\mathbf{K}$ has bandwidth m (meaning that all nonzero entries are within m rows of the diagonal, so for us $m = 17$), then $\mathbf{L}$ also has bandwidth m [1, p. 154]. After this step, a new energy $E(\mathbf{y}) = \frac{1}{2}\mathbf{y}^T\mathbf{y}$ with $\mathbf{y} = \mathbf{L}^T(\mathbf{w} - \hat{\mathbf{w}})$ yields a probability density function on $\mathbf{y}$ that is the product of uncoupled univariate normal distributions:

$$p_{\mathbf{y}}(\mathbf{y}) = \prod_{i=1}^{12n-6} \left(\frac{\beta}{2\pi}\right)^{\frac{1}{2}} e^{-\frac{\beta}{2}\mathrm{y}_i^2}. \tag{S.1}$$

To make a single draw $\mathbf{y}$ from this distribution, each component y_i is taken as a random number from the normal distribution with mean 0 and standard deviation $\beta^{-\frac{1}{2}}$. Note that units of the stiffness matrix $\mathbf{K}$ in the cgDNA model are such that $\beta = 1$. For the sake of efficiency uniform deviates are generated using the `xorshift1024*` implementation[2] of the `xorshift` algorithm [2] and are subsequently converted to normal deviates using the `ZIGNOR` implementation[3] of the Ziggurat algorithm [3].

The draw of the internal coordinates $\mathbf{w}$ corresponding to $\mathbf{y}$ is obtained from the equation $\mathbf{y} = \mathbf{L}^T(\mathbf{w} - \hat{\mathbf{w}})$ by solving $\mathbf{L}^T\mathbf{z} = \mathbf{y}$ for $\mathbf{z}$ (taking advantage of the upper triangular, banded structure of $\mathbf{L}$ using an appropriate solver from LAPACK [4]) and then setting $\mathbf{w} = \mathbf{z} + \hat{\mathbf{w}}$.

An alternative approach to obtain direct sampling would involve a spectral decomposition of $\mathbf{K}$ in place of the Cholesky factorisation, i.e.

$$\mathbf{K} = \mathbf{P}\mathbf{D}\mathbf{P}^T, \tag{S.2}$$

with $\mathbf{P}$ orthogonal and $\mathbf{D}$ diagonal. Here a similar change of variable $\mathbf{y} = \mathbf{D}^{\frac{1}{2}}\mathbf{P}^T\mathbf{w}$ can be used so that $\mathbf{w} = \mathbf{P}\mathbf{D}^{-\frac{1}{2}}\mathbf{y}$. This has been successfully exploited by Czapla et al. [5] for the case where $\mathbf{K}$ is block diagonal. However in our

[1] see `http://lcvmwww.epfl.ch/cgDNA`
[2] `http://arxiv.org/abs/1404.0390`
[3] http://www.doornik.com/research/ziggurat.pdf

setting with a (potentially large) banded $\mathbf{K}$ that approach is significantly less efficient, since the matrix $\mathbf{P}\mathbf{D}^{\frac{1}{2}}$ would not be sparse, and a dense matrix-vector multiply must be carried out in the construction of each draw. To give an example a simulation calculating $\langle \mathbf{t}_i^{[0]} \cdot \mathbf{t}_0^{[0]} \rangle$ for 1 million configurations of the λ_3 sequence of length 300 bp using Cholesky decomposition takes just above 3 minutes, while using spectral decomposition the running time is around 2 hours.

S.2.3 Metropolis Monte Carlo sampling

As described in the main text, to sample the non-Gaussian distribution $(11)_2$ we use the Metropolis algorithm (see [6] for a treatment similar to that we use here). Given a prior configuration with internal-variable vector $\mathbf{w}$, we follow the direct Monte Carlo procedure from the previous section to generate a new draw $\mathbf{w}^*$ and accept or reject it as follows: if $J(\mathbf{w}^*) \geq J(\mathbf{w})$, we accept $\mathbf{w}^*$, whereas if $J(\mathbf{w}^*) < J(\mathbf{w})$ we accept $\mathbf{w}^*$ with probability $J(\mathbf{w}^*)/J(\mathbf{w})$ and otherwise reject it (in which case we append a new copy of $\mathbf{w}$ to our ensemble). This acceptance criterion is one way of ensuring the crucial property of *detailed balance*, which requires that

$$\begin{aligned} &\alpha(\mathbf{w} \to \mathbf{w}^*) P(\mathbf{w} \to \mathbf{w}^*) \tilde{p}_{\mathbf{w}}(\mathbf{w}) \\ &\quad = \alpha(\mathbf{w}^* \to \mathbf{w}) P(\mathbf{w}^* \to \mathbf{w}) \tilde{p}_{\mathbf{w}}(\mathbf{w}^*), \end{aligned} \tag{S.3}$$

where $\tilde{p}_{\mathbf{w}}$ is the probability density function $(11)_2$, $\alpha(\mathbf{y} \to \mathbf{z})$ is the conditional probability density in our Metropolis algorithm for choosing state $\mathbf{z}$ given prior state $\mathbf{y}$ (which in our scheme is independent of $\mathbf{y}$ and equals $p_{\mathbf{w}}(\mathbf{z})$ from $(11)_1$), and $P(\mathbf{y} \to \mathbf{z})$ is the probability in our Metropolis algorithm of accepting the new state $\mathbf{z}$ given a prior state $\mathbf{y}$ (which in our scheme is 1 if $J(\mathbf{z}) \geq J(\mathbf{y})$ and $J(\mathbf{z})/J(\mathbf{y})$ otherwise).

The efficiency of any Metropolis method depends strongly on the acceptance rate for the given move set, which can be punitively small. In the particular case of the pdf $(11)_2$ with the explicit choice (12) for J, and the cgDNA energy (10), the observed acceptance rates depend on the length of the simulated oligomers. For oligomers of 300 bp the acceptance rate is approximately 37%, which is perfectly acceptable. For oligomers 5 times as long (1500 bp – as used for computing the Flory persistence vectors) the acceptance rate drops to just under 5%, with a corresponding increase in the number of draws required to obtain convergence.

S.2.4 Rigid base pair marginals

We remark that many expectations of interest involve only the *inter* part of the configuration variable $\mathbf{w}$ so that the number of degrees of freedom can be reduced by one half by computing the marginal distribution for the inter variables. As the original distribution is Gaussian its marginals are also Gaussian, but the resulting marginal stiffness matrix is now dense, so that sparse computations can no longer be used. As a consequence a calculation of $\langle \mathbf{t}_0^{[0]} \cdot \mathbf{t}_i^{[0]} \rangle$ for 1 million configurations of the λ_3 fragment using the marginal distribution takes around 23 minutes, nearly 7 times slower than generating ensembles in the full $\mathbf{w}$ space and discarding all the *intra* variables.

S.2.5 Reconstruction of 3D shapes

The first step in calculating our observables is reconstructing a 3D shape of a molecule from a given internal coordinate vector $\mathbf{w}$ as detailed in [7]. As mentioned in the previous section, the calculation of tangent-tangent correlations, arclengths and Flory vectors require only the *inter* part of $\mathbf{w}$. As a result we only reconstruct base pair positions $\mathbf{r}_i$ and orientations $\mathbf{R}_i$, which takes only half the time of reconstructing a full 3D configuration of rigid bases. The reconstruction procedure, implemented by the *cgDNArecon* library, involves evaluating half rotations, composing rotations, applying rotations to vectors and adding vectors.

A careful numerical study of efficiency of different parametrisations of rotations (namely Cayley vectors, unit quaternions and rotation matrices) using the *algebra3d* library has been performed. An explicit half-rotation formula for unit quaternions proved to be 60% faster than a similar formula for Cayley vectors. (For rotation matrices, the analogous calculation would require, e.g., an iterative algorithm of computing the principal square root and so was not considered). As expected, for composition of rotations, quaternion multiplication was faster than matrix multiplication, with our observed difference being 20%. On the other hand, in the case of applying a rotation to a vector, the matrix-vector product was 5 times faster than a specialised quaternion multiplication. In fact the fastest way to apply a rotation given as unit quaternion to a vector was to convert the quaternion to a rotation matrix first (this takes only twice the time of the matrix-vector product). Efficiency of converting between all three parametrisations was also analysed. This suggested, for example, that the formula for computing a rotation matrix

for a given Cayley vector of [7] is two times slower than conversion of a Cayley vector to quaternion and subsequent conversion of the quaternion to a rotation matrix.

Considerations similar to the above suggested two approaches to the reconstruction procedure. The first one uses directly the Cayley vectors of the configuration variable $\mathbf{w}$ to calculate half rotations and converts to rotation matrices for all subsequent calculations. The other one, that finally proved to be 30% faster, begins with converting the Cayley vectors to quaternions, then computes half rotations using quaternions, and finally converts quaternions to matrices when rotations need to be applied to vectors.

S.2.6 Remarks on parallelisation

We first note that in *cgDNAmc* pseudo-random numbers are generated sequentially to ensure reproducibility of results. Also the reconstruction procedure is inherently sequential. The conversion of the decoupled normal deviates $\mathbf{y}$ to an internal coordinate vector $\mathbf{w}$ depends on the underlying LAPACK routine, that might already be optimized to use available multiple cores, but the *cgDNAmc* code has no other explicit parallelisation. In part this is because each configuration can be generated and analysed independently of all others, so that the suggested solution for generating large ensembles is to run multiple independent simulations at the same time, with a different seed for the pseudo-random number generator in each instance. By linearity, expectations from multiple runs can be aggregated as a weighted average with weights proportional to the number of configurations generated in each independent run. As an example we achieved a 2.4 speed up in this way by running four independent threads on a single laptop.

S.2.7 Run-times of key steps of algorithm

A simple profile of run times for the key steps of a simulation that calculates five expectations using 1 million configurations of the 300 bp λ_3 oligomer is:

Operation	Run time [s]	% of simulation
Generation of $\mathbf{y}$	59.08	12.66%
Transformation to $\mathbf{w}$	74.75	16.02%
Shape reconstruction	41.41	8.87%
Calculating $\langle \mathbf{t}_0^{[0]} \cdot \mathbf{t}_i^{[0]} \rangle$	6.63	1.42%
Calculating $\langle \mathbf{t}_0^{[11]} \cdot \mathbf{t}_i^{[11]} \rangle$	273.28	58.55%
Calculating $\langle s_i^{[1]} \rangle$	3.72	0.80%
Calculating $\langle s_i^{[11]} \rangle$	0.61	0.13%
Calculating Flory vecs	6.12	1.30%
Other	1.14	0.24%
Entire simulation	466.74	100.00%

The time necessary to evaluate most of the expectations is a negligible fraction of the total, except for the generalized-chord expectation $\langle \mathbf{t}_0^{[11]} \cdot \mathbf{t}_i^{[11]} \rangle$, where the computation of the principal eigenvector of the local gyration matrix is quite costly.

S.3 DNA sequences

S.3.1 λ phage genome

Five fragments of length 300 bp drawn from the λ phage genome of Sanger et al. [8]. The full sequence is available online[4]. A single repeat was used for ℓ_p computations, 5 repeats for ℓ_F.

λ_1 (base pairs 25201 – 25500)

```
TTGTAGGCTC AAGAGGGTGT GTCCTGTCGT AGGTAAATAA CTGACCTGTC
GAGCTTAATA TTCTATATTG TTGTTCTTTC TGCAAAAAAG TGGGGAAGTG
AGTAATGAAA TTATTTCTAA CATTTATCTG CATCATACCT TCCGAGCATT
TATTAAGCAT TTCGCTATAA GTTCTCGCTG GAAGAGGTAG TTTTTTCATT
```

[4] http://www.ncbi.nlm.nih.gov/nuccore/215104

```
GTACTTTACC TTCATCTCTG TTCATTATCA TCGCTTTTAA AACGGTTCGA
CCTTCTAATC CTATCTGACC ATTATAATTT TTTAGAATGG TTTCATAAGA
```

λ_2 (base pairs 21901 – 22200)

```
CGTTAACGCT GCGGGTAACG CGGAAAACAC CGTCAAAAAC ATTGCATTTA
ACTATATTGT GAGGCTTGCA TAATGGCATT CAGAATGAGT GAACAACCAC
GGACCATAAA AATTTATAAT CTGCTGGCCG GAACTAATGA ATTTATTGGT
GAAGGTGACG CATATATTCC GCCTCATACC GGTCTGCCTG CAAACAGTAC
CGATATTGCA CCGCCAGATA TTCCGGCTGG CTTTGTGGCT GTTTTCAACA
GTGATGAGGC ATCGTGGCAT CTCGTTGAAG ACCATCGGGG TAAAACCGTC
```

λ_3 (base pairs 36901 – 37200)

```
TAGAGCGATT TATCTTCTGA ACCAGACTCT TGTCATTTGT TTTGGTAAAG
AGAAAAGTTT TTCCATCGAT TTTATGAATA TACAAATAAT TGGAGCCAAC
CTGCAGGTGA TGATTATCAG CCAGCAGAGA ATTAAGGAAA ACAGACAGGT
TTATTGAGCG CTTATCTTTC CCTTTATTTT TGCTGCGGTA AGTCGCATAA
AAACCATTCT TCATAATTCA ATCCATTTAC TATGTTATGT TCTGAGGGGA
GTGAAAATTC CCCTAATTCG ATGAAGATTC TTGCTCAATT GTTATCAGCT
```

λ_4 (base pairs 24301 – 24600)

```
CTATGACTGT ACGCCACTGT CCCTAGGACT GCTATGTGCC GGAGCGGACA
TTACAAACGT CCTTCTCGGT GCATGCCACT GTTGCCAATG ACCTGCCTAG
GAATTGGTTA GCAAGTTACT ACCGGATTTT GTAAAAACAG CCCTCCTCAT
ATAAAAAGTA TTCGTTCACT TCCGATAAGC GTCGTAATTT TCTATCTTTC
ATCATATTCT AGATCCCTCT GAAAAAATCT TCCGAGTTTG CTAGGCACTG
ATACATAACT CTTTTCCAAT AATTGGGGAA GTCATTCAAA TCTATAATAG
```

λ_5 (base pairs 37801 – 38100)

```
CCTGACTGCC CCATCCCCAT CTTGTCTGCG ACAGATTCCT GGGATAAGCC
AAGTTCATTT TTCTTTTTTT CATAAATTGC TTTAAGGCGA CGTGCGTCCT
CAAGCTGCTC TTGTGTTAAT GGTTTCTTTT TTGTGCTCAT ACGTTAAATC
TATCACCGCA AGGGATAAAT ATCTAACACC GTGCGTGTTG ACTATTTTAC
CTCTGGCGGT GATAATGGTT GCATGTACTA AGGAGGTTGT ATGGAACAAC
GCATAACCCT GAAAGATTAT GCAATGCGCT TTGGGCAAAC CAAGACAGCT
```

S.3.2 Virstedt et al. sequences

Sequences used in the experimental study by Virstedt et al. [9]. A single repeat was used for ℓ_p computations, 8 for ℓ_F.

γ_1 (CA) – 170 base pairs

```
GAGGATTCCT GGGAAAACCC TGGTACACAC ACACCACATC ATGCATACAC
ACACATCATG CATGCATACA CACATACATA CACATACTAA CACATACACT
CACACACACG CCACAAATTA TGCATGCATA CACACATGCA CGCACACACA
CAGGAAACAG CTCGGTCCTC
```

γ_2 (CAG) – 181 base pairs

```
GAGGATTCCT GGGAAAACCC TGGCGAGCAG CAGCAGCAAC AGTAGTAGAA
GCAGCAGCAC TAACGACAGC AGCAGCAGTA GCAGTAATAG AAGCAGCAGC
AGCAGCAGTA GCAGTAGCAG CAGCAGCAGC AGCAATAACA ACAACAGCAG
CAGCAGTCAC ACAGGAAACA GCTCGGTCCT C
```

γ_3 (NoSeq) – **195 base pairs**

```
GAGGATTCCT GGGAAAACCC TGGCGCAAGA CCGAGTTACT AAACAGGACT
ATTACTGCCA CGCCAATTGT AGCGCGCAGC CACGTCTCTG CTCACCACTA
TCCTCTTGTT GACGCTATTG CTACTATCGC ATCCCGCTTA GCTATACCTA
CTGATGCTCA ATTACCCGCC TCACACAGGA AACAGCTCGG TCCTC
```

γ_4 (TATA) – **176 base pairs**

```
GAGGATTCCT GGGAAAACCC TGGCGAGGTC TATAAGCGTC TATAAGCGTC
TATGAACGTC TATAAACGTC TATAAACGCC TATAAACGCC TATAAACGCC
TATACAAGCC TATAAACGCC TATACACGTC TATGCACGAC TATACACGTC
TTCACACAGG AAACAGCTCG GTCCTC
```

S.3.3 Bednar et al. [10] γ_5

A sequence designed to be intrinsically straight. 9-15 repeats were used for ℓ_p and ℓ_w computations, 75 for ℓ_F.

```
ATCTAATCTA ACACAACACA
```

S.3.4 Kahn and Crothers [11] c11t15/γ_6

An intrinsically bent sequence with phased A-tracts originally used for minicircle experiments. 2 repeats were used for ℓ_p computations, 10 for ℓ_F.

```
GATGAATTCA CGGATCCGGT TTTTTGCCCG TTTTTTGCCG TTTTTTGCCC
GTTTTTTGCC GTTTTTTGCC CGTTTTTTCC GGATCCGTAC AGGAATTCTA
GACCTAGGGT GCCTAATGAG TGAGCTAACT CACATTAATT GCGTTGCGCC
ATGGAATC
```

S.3.5 Geggier and Vologodskii [12] sequences

Sequences used in the experimental study by Geggier and Vologodskii as provided in their Supplementary Information. A single copy was used for ℓ_p computations.

ACAT – **201 base pairs**

```
AGCTTACACA TATATACACA TACATATACA CACACATATA CTGCAGACAT
ACACATATAT ACACATACAT ACACACATAC ATATATATAC ACACACATAT
ACATACATAT ACATACATAC ATATATACAC ACATACATAC ACATATATAT
ACACATACAC ATACATACAC ACATATACAT ATACATACAC ATATACACAT
A
```

ACCAGG – **201 base pairs**

```
AGCTTACCAG GAGGACCACC AGGACCACCA CCAGGAGGAG CTGCAGACCA
GGACCACCAG GAGGACCAGG AGGAGGACCA CCACCAGGAC CACCAGGACC
AGGAGGAGGA CCACCAGGAC CACCAGGAGG AGGACCACCA GGACCAGGAG
GACCACCACC AGGAGGAGGA CCACCAGGAC CAGGAGGACC AGGACCACCA
A
```

ACGAGC – **199 base pairs**

```
AGCTTAGCAC GACGAGCAGC ACGAGCAGCA GCACGACGCT GCAGAGCAGC
ACGACGACGA GCACGAGCAG CACGACGAGC ACGAGCAGCA GCACGAGCAC
GACGAGCAGC ACGACGAGCA GCACGACGAG CACGACGACG AGCAGCAGCA
CGAGCAGCAC GAGCACGACG ACGAGCACGA GCAGCACGAG CAGCACGAA
```

AGAT – **200 base pairs**

```
AGCTTAGAGA TATATAGAGA TAGATATAGA GAGAGATATC TGCAGAGATA
GAGATATATA GAGATAGATA GAGAGATAGA TATATATAGA GAGAGATATA
GATAGATATA GATAGATAGA TATATAGAGA GATAGATAGA GATATATATA
GAGATAGAGA TAGATAGAGA GATATAGATA TAGATAGAGA TATAGAGATA
```

AGC – **199 base pairs**

```
AGCTTAGCAG CAGCAGCAGC TAGCAGCAGC AGCCTGCAGA GCAGCAGCAG
CAGCTAGCAG CAGCAGCAGC AAGCAGCAGC AGCAGCGAGC AGCAGCAGCA
GCTAGCAGCA GCAGCAGCAA GCAGCAGCAG CAGCGAGCAG CAGCAGCAGC
TAGCAGCAGC AGCAGCAAGC AGCAGCAGCA GCGAGCAGCA GCAGCAGCA
```

CAA – **200 base pairs**

```
AGCTTACAAC AACAACAACC TGCAGAACCA ACAACAAGCA CAACAACAAC
AACAACAAGA ACAACAACAA CAACAACCAA CAACAACAAC AACAACCAAC
AACAACAAGC AACAACAACA ACAACAACCA ACAACAACAA CCAACAACAA
CAACCAACAA CAACAACAGC AACAACAACA ACAACAAGAA CAACAAGAAA
```

CAACTT – **198 base pairs**

```
AGCTTCAACT TCTTCAACAA CAACTTCTTC TTCAACTCTG CAGCAACTTC
TTCAACAACT TCAACTTCTT CAACAACAAC TTCTTCTTCA ACAACTTCAA
CAACTTCTTC AACTTCAACA ACTTCTTCAA CTTCAACTTC TTCAACAACA
ACTTCTTCAA CTTCTTCAAC AACTTCAACT TCAACAACTT CTTCAACA
```

CAGT – **200 base pairs**

```
AGCTTCAGTC AGTCAGTCTG ACAGTCAGTC AGTCAGTCAG TCAGTCTGCA
GCAGTCAGTC AGTCAGTCAG TCAGTCAGTC TGACAGTCAG TCAGTCAGTC
AGTCAGTCAG TCAGTCTGAC AGTCAGTCAG TCAGTCAGTC AGTCAGTCAG
TCAGTCTGAC AGTCAGTCAG TCAGTCAGTC TGACAGTCAG TCAGTCAGTA
```

CATCTA – **200 base pairs**

```
AGCTTCATCT ACTACATCAT CATCTACTAC ATCTACATCC TGCAGCATCA
TCTACTACAT CTACTACTAC ATCATCATCT ACATCTACTA CATCATCTAC
TACATCATCT ACTACATCAT CTACATCTAC ATCTACTACA TCATCTACTA
CATCATCATC TACTACATCT ACTACATCTA CATCATCTAC TACATCATCA
```

HPL1 – **198 base pairs**

```
AGCTTCGATT GCGCATTCGA TTGGAGTCTC CGCTGCCATT GCATTCTGCA
GCGATTCGGC ATTCGATTCG CGCATTCGAT TCGCATTCGA TTCGGCATTC
GATTCGGCAT TCGATTCGCA TTCGATTCAT TCATTCGATT CGGCATTCGA
TTCGGCATTC GATTGCGATT GCATTCGGCA TTCGATTCGG CGATTCAA
```

LPL1 – **200 base pairs**

```
AGCTTTAGTA GCCTAGTAGC CTAGAGTCTC CGCTGCCATT GCCCTACCTG
CAGTAGTAGC CTAGTAGCCT AGTAGCCTAG TAGCCTAGTA GCCTAGTAGC
CTAGTAGCCT AGTAGCCTAG TAGCCTAGTA GCCTAGTAGC CTAGTAGCCT
AGTAGCCTAG TAGCCTAGTA GCCTAGTAGC CTAGTAGCCT AGTAGACTAA
```

HPL2 – **198 base pairs**

```
AGCTTACGAC GAACGACGAC GAACGACGAA CGAACGACGA ACGCTGCAGA
CGACGAACGA CGAACGACGA CGAACGAACG ACGACGAACG ACGAACGACG
ACGAACGACG ACGAACGACG AGACGAACGA ACGACGACGA ACGACGAACG
ACGACGAACG ACGAACGACG AACGACGACG AACGAACGAC GAACGACA
```

LPL2 – **201 base pairs**

```
AGCTTGCATA GGCATTAGCC ATGCATAGGC ATATGGCATT AGGCACTGCA
GGGCCATAGG CATGCATAGG CATAGGCCAT GGCATAGGCA TTAGGCATGC
ATAGGCATAG GCATGGCATA GGCATTAGGC ATGCATAGGC ATAGGCATGG
CATAGGCATT AGGCATGCAT AGGCATGGCA TAGGCCATGG CATAGGCATT
A
```

SG1 – **199 base pairs**

```
AGCTTAGGAC TACGAACGCT AGCTTAGCTA CCAGCGAGTA CACTGCAGCA
GCAGCTAGCT AGCGCGATGC CCAGCTGAGA TCGACGATCG ATGGCGATTA
TCAGCTAGCA GCTAGCGATC GACGCGCGAT GCGCAGCTGA GCTAGCTGAT
CAGCTTCAGC TGACGTCAGC TGAGAGCTGA CCACCGTAGA GTCGATCGA
```

λ_6 – **205 base pairs**

```
AGCTTCTCCT TTGATGCGAA TGCCAGCGTC AGACATCATA TGCAGATACT
CACCTGCATC CTGAACCCAT TGACCTCCAA CCCCGTAATA GCGATGCGTA
ATGATGTCGA TAGTTACTAA CGGGTCTTGT TCGATTAACT GCCGCAGAAA
CTCTTCCAGG TCACCAGTGC AGTGCTTGAT AACAGGAGTC TTCCCAGGAT
GGCGA
```

γ_7 (IS) – **211 base pairs**

```
CTAGAAGCTT ACTCGACTCG AGCCTAGCCT ATGACATGAC ACGTTACGTT
AGTCGAGTCG ATCAGATCAG ACGCTACGCT AGCTGAGCTG ACTGTACTGT
ATGCAATGCA ACCTCACCTC AGGACAGGAC ACGTGACGTG ATGCTATGCT
ACCAGACCAG CTGCACTGCA GACTGGACTG ACGCTACGCT ATCGCATCGC
AGATGAGATG A
```

S.4 Supplementary Figures

S.4.1 Sequence is significant—some specific cases

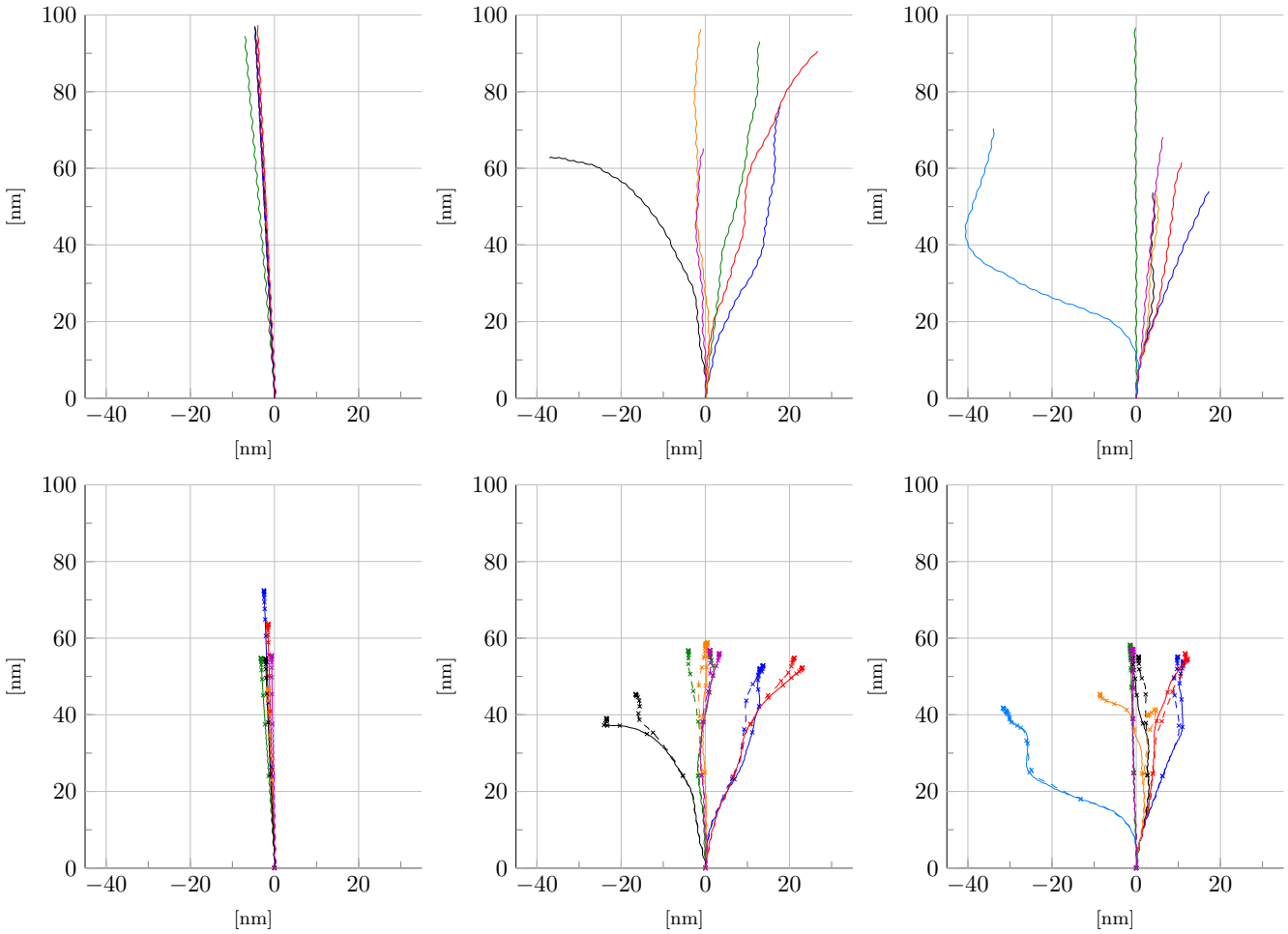

Figure S1: *Ground state configurations and Flory persistence vectors for various DNA sequences (an interactive version of Figure 4 of the main text). The columns show: (left) the six distinct poly-dinucleotide sequences, (middle) the six selected λ-phage fragments λ_j, and (right) the seven sequences γ_j. The first row of panels shows visualizations of the shapes of* cgDNA *ground state configurations, while the second row shows plots of Flory persistence vectors $(1)_2$ for the Gaussian $(11)_1$ (solid) and perturbed $(11)_2$ (dashed) ensembles. (All six panels are U3D, so that with the appropriate viewer, e.g. Acrobat Reader V7 or higher, they can be interactively rotated and magnified.)*

S.4.2 Sensitivity to Jacobian perturbation

We used the two sequences poly(G) and λ_3 to explore the sensitivity of the persistence length values $\ell_p(\mathbb{S})$ on the inclusion of the Jacobian factor from Eq. (12) in the probability density function, as seen in Eq. $(11)_2$. Results are presented in Fig. S2. Some difference is perceptible, including a difference in the period of the small oscillations in the case of poly(G), but the magnitude of these effects is rather small at these length scales.

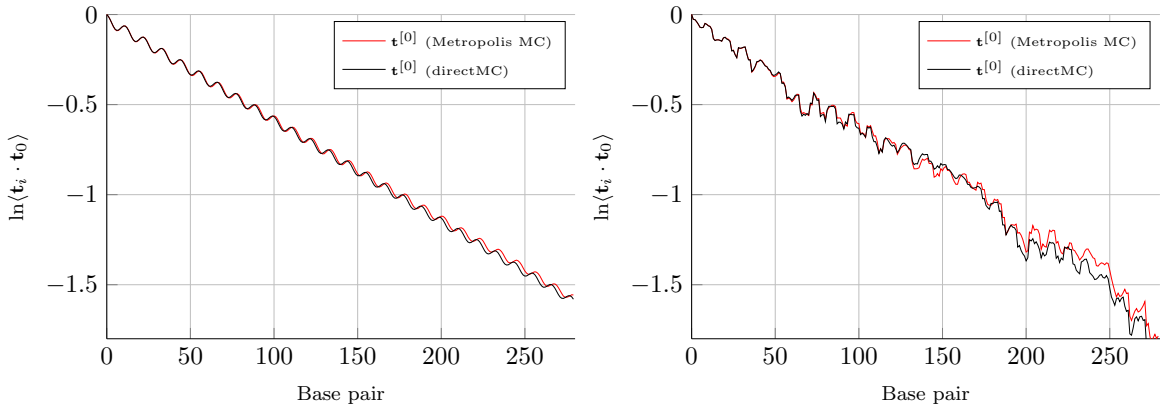

Figure S2: *Sensitivity of tangent-tangent correlation data to inclusion of Jacobian factor. Direct Monte Carlo simulation (which does not use the Jacobian) in black, Metropolis Monte Carlo (which incorporates Jacobian) in red; left panel is for 300 bp poly(G) fragment (*173 *bp – direct,* 175 *bp – Metropolis) and right panel is for* λ_3 *(*162 *bp – direct,* 167 *bp – Metropolis). In each case* 10 *bp were excluded at each end.*

S.4.3 Monte Carlo convergence

In the main document, we report two types of convergence results for the Flory persistence vector, first that 10^5 MC draws is a sufficiently large number of samples, and second that 1.5 Kbp is a sufficiently long sample to yield an overall standard error of less than 0.5 nm. For the apparent persistence length $\ell_p(\mathbb{S})$, we similarly report that 10^5 MC samples are sufficient for an accuracy of 0.5 nm (using $j = k = 11$). These convergence conclusions are illustrated in Fig. S3.

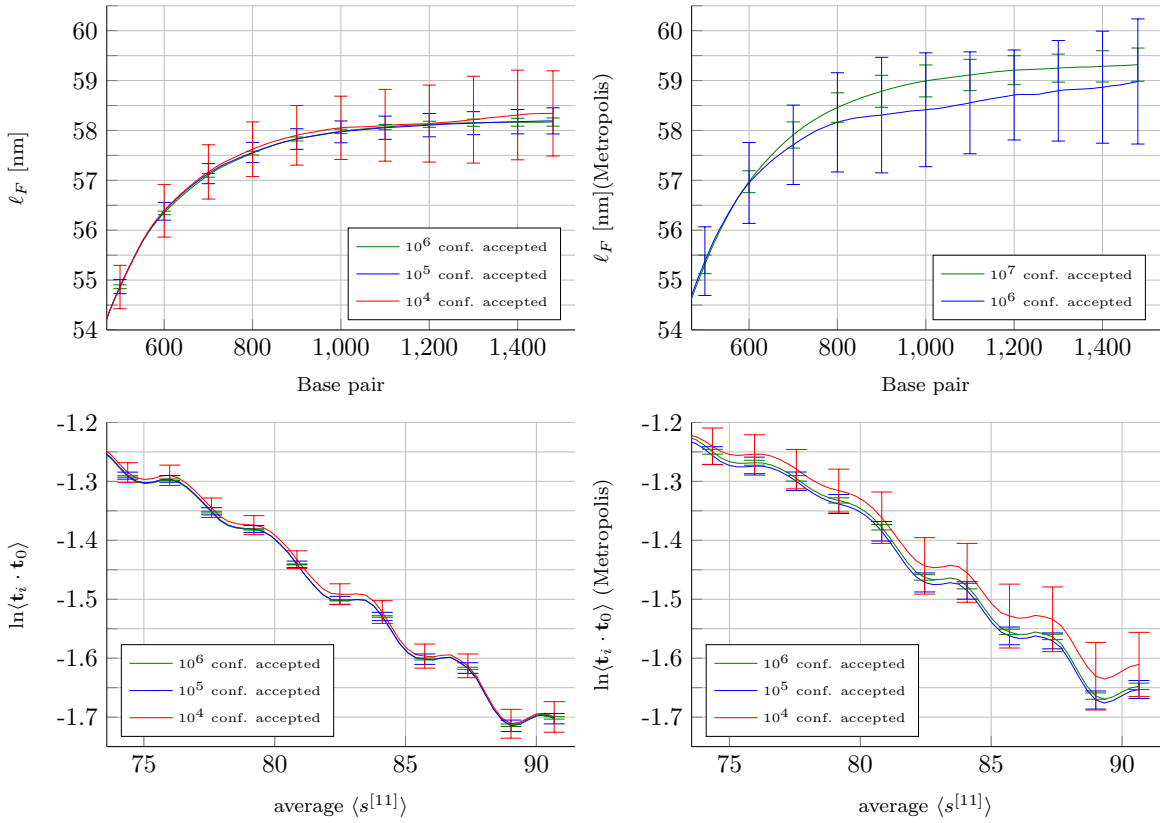

Figure S3: *Examples of convergence of direct MC sampling (left column) and Metropolis MC (right column) for λ_3. In the top two panels the curves show the norm of the Flory vector (averaged over MC samples of different sizes) plotted against base-pair number; in all cases, the asymptotic values appears to have been reached by 1.5 Kbp (five repeats of λ_3). The error bars give the standard error obtained for ten independent MC runs plotted every 100 bp; in the case of direct MC with 10^5 MC samples, the error bars shrink below ± 0.5 nm, however for Metropolis MC as many as 3×10^6 accepted configurations (with acceptance rate of 4%) are required for the same accuracy. In the bottom two panels, we show the last 50 bp of the tangent-tangent correlation plot relevant for computing ℓ_p of a single repeat of λ_3 with coarse grain parameters $[j, k] = [11, 11]$ (averaged over MC samples of different sizes). The error bars, (plotted every 5 bp) give the standard error obtained for ten independent MC runs. The values of ℓ_p extracted from these tangent-tangent plots are 56.5 nm (for direct MC) and 57.5 nm (for Metropolis MC with 37% acceptance rate), both with at least 0.5 nm accuracy for 10^5 or more accepted MC samples.*

S.4.4 Coarse Graining sensitivity

This section provides data to justify assertions made in the main text concerning the dependence of $\ell_p^{[j,k]}$ on the choice of coarse graining parameters $[j, k]$. Figure S4 illustrates some of the tangent-tangent correlation data which is fit to extract $\ell_p^{[j,k]}$. Numerical values of persistence lengths for further cases are presented in Table S2.

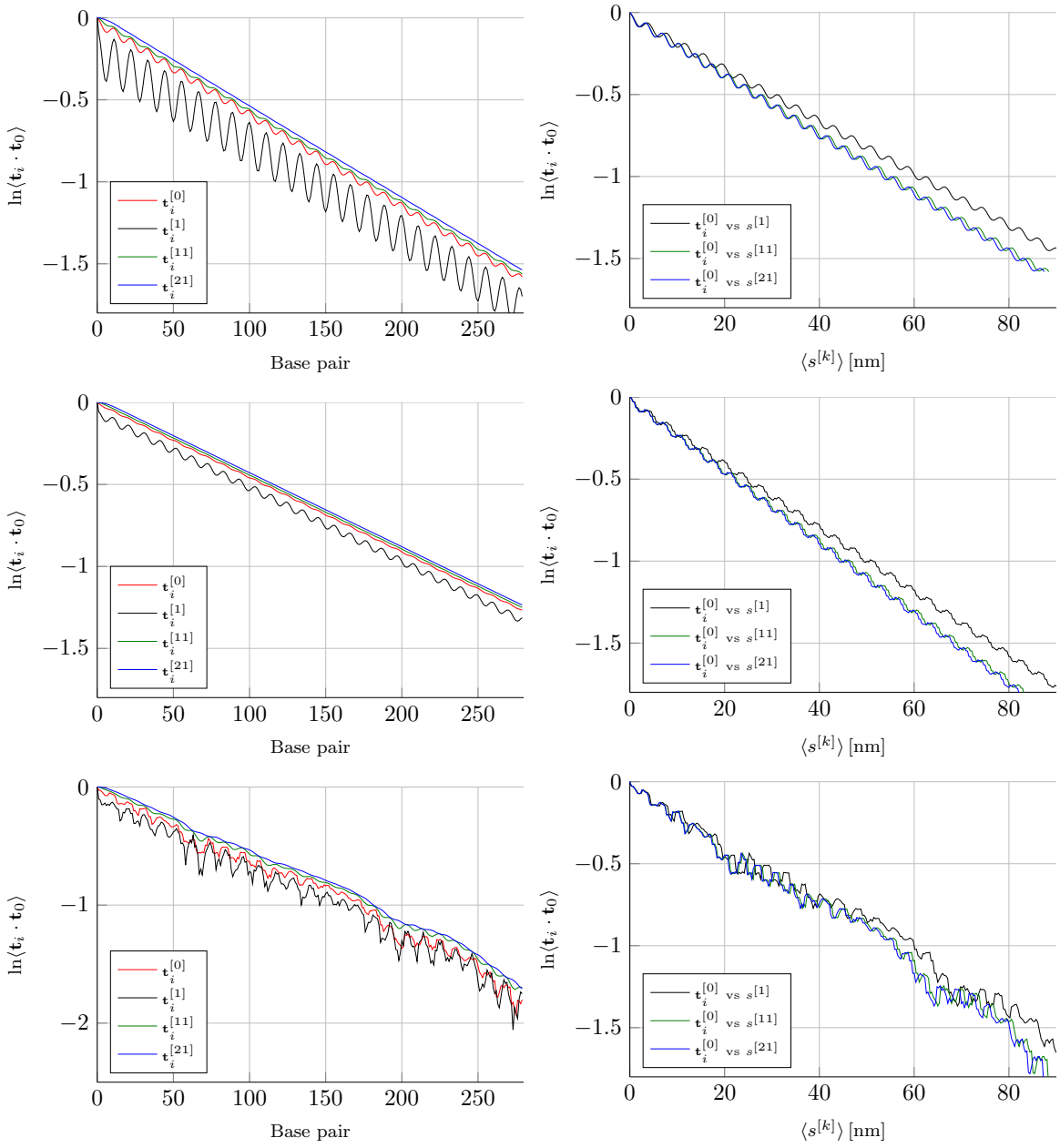

Figure S4: *Left column. The data* $\ln\left\langle \mathbf{t}_i^{[j]} \cdot \mathbf{t}_0^{[j]} \right\rangle$ *for fitting* $\ell_p^{[j,0]}(\mathbb{S})$ *with different coarse-graining approximations of tangents. Each panel shows the cases* $j = 0, 1, 11, 21$ *for each of the three sequences poly(G) (top), poly(A) (middle),* λ_3 *(bottom). The cases* $j = 0, 11, 21$ *yield very similar plots, whereas* $j = 1$ *shows some significant deviation with much larger oscillation and an overall displacement downward, meaning that in the estimation of persistence length the gradient of the linear best fit is quite sensitive to whether or not the line is assumed to pass through the origin. The plots for* $j = 0$ *and* $j = 11$ *exhibit oscillations of roughly the same magnitude, presumably reflecting an alignment of the base pairs with the local axis of the local helical structure. In addition, the initial behaviour for the first few base pairs is notably different for the choices* $k = 0, 1, 11, 21$ *(which is entirely reasonable given the block structure of the stiffness matrix in the* cgDNA *model) which leads to the consistent (approximate) ordering* $1 < 0 < 11 < 21$*. Right column. The data for fitting the dimensional persistence lengths* $\ell_p^{[0,k]}(\mathbb{S})$ *for different coarse-graining choices of arc length. Each panel shows the cases* $k = 1, 11, 21$ *for each of the three sequences poly G (top), poly A (middle) and* λ_3 *(bottom). For poly(G) there is a difference of* 6.4 *nm between* $k = 1$ *and* $k = 11$ *while for poly(A) the same difference is only* 3.0 *nm.*

S.4.5 Sequence-dependence of persistence lengths

Figure S5 below is the analogue of Fig. 3, which shows histograms of ℓ_F (in nm) and $\ell_p^{[0,0]}$, and $\ell_d^{[0,0]}$ (in bp units), but now for $\ell_p^{[11,11]}$ and $\ell_d^{[11,11]}$ in nm.

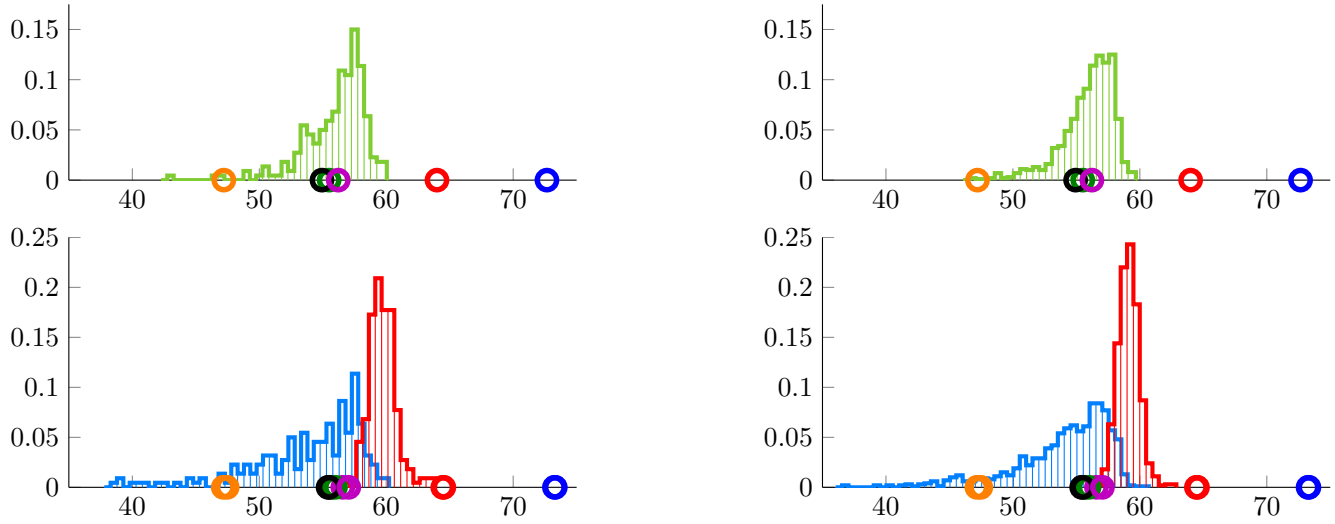

Figure S5: *Normalised histograms of persistence lengths ℓ_F in green, $\ell_p^{[11,11]}$ in blue, $\ell_d^{[11,11]}$ in red (all in nm units) for 220 bp fragments from λ-phage (left) and with random sequence (right). In addition, in each panel, the associated persistence lengths for the six distinct poly(dinucleotide) sequences are marked with coloured circles, with two circles per sequence as ℓ_p and ℓ_d almost coincide for these almost straight sequences. The harmonic means of $\ell_F(\mathbb{S}_j)$ for the λ and random ensembles are respectively 55.7 nm and 55.6 nm, of $\ell_d(\mathbb{S}_j)$ 59.5 nm and 58.8 nm, and of $\ell_p(\mathbb{S}_j)$ 53.2 nm and 53.5 nm.*

S.4.6 Sequence-averaged persistence lengths

Figure S6 shows the tangent-tangent correlation plots that were used to compute $\bar{\ell}_d$ and $\bar{\ell}_p$.

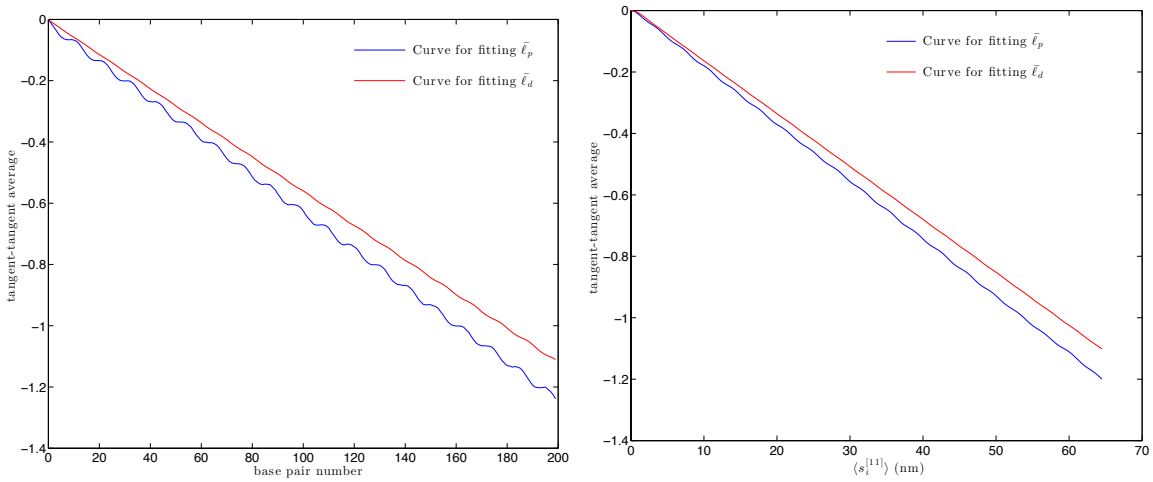

Figure S6: *Tangent-tangent correlation plots used for extracting* $\bar{\ell}_p{}^{[0,0]}$ *and* $\bar{\ell}_d{}^{[0,0]}$ *in bp units (left panel) and* $\bar{\ell}_p{}^{[11,11]}$ *and* $\bar{\ell}_d{}^{[11,11]}$ *in nm units (right panel). As per the definitions of these quantities, each blue curve is the log of the average over sequence and over MC samples of* $\mathbf{t}_i^{[j]} \cdot \mathbf{t}_0^{[j]}$*, while each red curve is the log of the average over sequence and MC samples of the ratio* $(\mathbf{t}_i^{[j]} \cdot \mathbf{t}_0^{[j]})/(\hat{\mathbf{t}}_i^{[j]} \cdot \hat{\mathbf{t}}_0^{[j]})$ *(*$j = 0$ *in left panel,* $j = 11$ *in right panel). In the left panel, the factorization that involves dividing by the intrinsic shape term greatly reduces the oscillations apparent in the blue curve. In the right panel, the use of a coarse grain arclength also reduces the oscillations in the blue curve relative to the blue curve in the left panel.*

S.4.7 Some tangent-tangent simulated correlation plots for sequences with experimental data

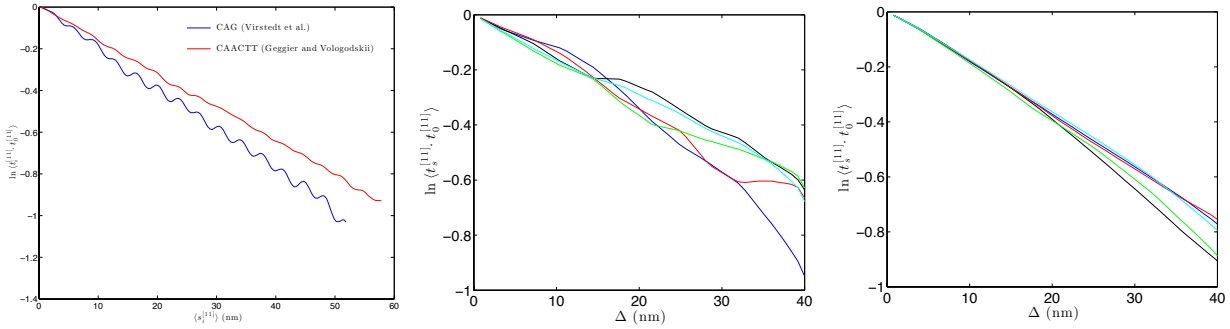

Figure S7: *Tangent-tangent correlation data used for extracting* ${\ell_p}^{[11,11]}$ *and* ${\ell_w}^{[11,11]}$. *Left panel: plots for a sequence with relatively high* ℓ_p *(CAACTT from Geggier and Vologodskii, red) and one with relatively low* ℓ_p *(CAG from Virstedt* et al., *blue). Middle and right panels: Five plots each for extracting* ℓ_w *for Bednar* et al.*'s straight molecule (middle) and* λ*-phage measurement (right). Each of the five plots in each panel represents an average over 25 MC samples and over all possible* 1*-nm-shifted windows of each given width* Δ *up to* 40 *nm (avoiding the first or last* 15 *bp). Note the wide variation in curve shape and best-fit slope* $-1/\ell_w$ *among the five curves in the middle panel, corresponding to the relatively large uncertainty (*$\pm$15 *nm) in Bednar* et al.*'s reported value of* ℓ_w. *The variation is less prominent in the right panel since there are more fragments (*37 *as compared to* 25*) and more windows per fragment (as the* λ *fragments are* 300 *bp whereas the straight molecule is* 180 *bp). For our reported values of* ℓ_w *(*58 *nm and* 52 *nm respectively), we computed* 1,000 *such curves and averaged the resulting values of* ℓ_w*; Bednar* et al.*'s reported values would seem to be the result of analysing a single such curve, and for a single window size of* 40 *nm.*

S.5 Tables of Numerical Data

Table S1 provides the numerical values of the persistence lengths extracted from the plots of simulated expectations presented in Figures 4 and 5 in the main text, along with the numerical values of the points in the scatter plot of computed versus experimentally observed persistence lengths shown in Figure 7. Table S2 quantifies the sensitivity of persistence lengths to coarse-grain choices $[j, k]$ in arc-length and tangent fit in the cases of four sequences. Tables S3 and S4 provide the MD and MC numerical simulation data used to compare fits of persistence lengths for short fragments of the six distinct poly-dimer sequences, cf. Figure 6 and Table 4 in the main text.

Group	Molecule	$\ell_p^{[0,0]}$ (bp)	$\ell_d^{[0,0]}$ (bp)	ℓ_F (nm)	ℓ_{F_J} (nm)	$\ell_p^{[11,11]}$ (nm)	ℓ_p^{expt} (nm)
Poly-dinucleotides	poly(AA)	219	221	72.7	70.2	73.5	50.4
	poly(GG)	173	178	56.2	55.7	56.0	41.7
	poly(TA)	146	148	47.2	46.9	47.0	42.7
	poly(AC)	169	173	55.5	56.6	55.4	50.7
	poly(AG)	192	194	64.0	64.5	64.5	52.6
	poly(CG)	166	168	54.9	54.0	55.3	49.7
fragments of λ	λ_1	99	231	48.4	52.8		
	λ_2	152	184	55.3	55.9		
	λ_3	162	186	58.3	59.0		
	λ_4	171	181	56.2	57.0		
	λ_5	178	181	59.1	58.9		
	λ_6	168	178	57.4	57.6	56.9	48.0
Virstedt *et al* [9]	γ_1	166	172	54.9	57.0	54.8	45.5
	γ_2	155	176	56.0	56.1	51.5	41.7
	γ_3	169	176	56.1	56.4	56.3	50.5
	γ_4	153	172	48.3	42.9	51.5	45.5
Bednar *et al* [10]	γ_5	176	179	58.2	58.7	58.2	82
	Avg of λ					52	45
Kahn & Crothers [11]	γ_6	134	187	51.5	52.5		
Geggier & Vologodskii [12]	γ_7	172	175	57.2	57.5	57.4	49.5
	ACAT					51.4	46.0
	ACCAGG					54.2	47.5
	ACGAGC					57.9	51.0
	AGC					57.1	47.0
	CAA					60.0	50.0
	CAACTT					62.5	51.0
	AGAT					51.5	47.0
	CAGT					60.0	51.5
	LPL1					53.7	48.0
	CATCTA					53.7	49.0
	HPL1					58.6	48.5
	LPL2					51.5	45.5
	HPL2					57.9	54.0
	SG1					55.5	48.5

Table S1: *Numerical values of persistence lengths: Columns* 3–6 *are the persistence lengths derived from the data plotted in Figures 4 and 5 of the main text (where ℓ_{F_J} denotes the Flory persistence length computed from the distribution including the Jacobian factor). Columns* 7 *and* 8 *are data used in the comparison between simulation and experimental results shown in Figure 7 of the main text (experimental results taken from citations indicated in each section, with experimental results for poly-dinucleotides and λ_6 from [12], see main text).*

poly(AA)

$\ell_p^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	219	74.6	71.6	71.0
$\mathbf{t}^{[1]}$	205	69.7	66.9	66.3
$\mathbf{t}^{[11]}$	224	76.1	73.0	72.4
$\mathbf{t}^{[21]}$	228	77.4	74.3	73.7

poly(AA)

$\ell_d^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	221	75.0	72.0	71.4
$\mathbf{t}^{[1]}$	210	71.6	68.7	68.1
$\mathbf{t}^{[11]}$	224	76.1	73.1	72.5
$\mathbf{t}^{[21]}$	228	77.4	74.3	73.7

poly(AT)

$\ell_p^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	146	50.7	45.9	45.4
$\mathbf{t}^{[1]}$	129	44.9	40.7	40.2
$\mathbf{t}^{[11]}$	149	51.7	46.9	46.3
$\mathbf{t}^{[21]}$	153	53.0	48.1	47.5

poly(AT)

$\ell_d^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	148	51.5	46.7	46.1
$\mathbf{t}^{[1]}$	143	49.8	45.1	44.5
$\mathbf{t}^{[11]}$	150	52.1	47.2	46.6
$\mathbf{t}^{[21]}$	153	53.1	48.1	47.5

poly(GG)

$\ell_p^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	173	61.3	54.9	54.3
$\mathbf{t}^{[1]}$	149	52.7	47.2	46.7
$\mathbf{t}^{[11]}$	178	62.8	56.3	55.6
$\mathbf{t}^{[21]}$	183	64.6	57.9	57.2

poly(GG)

$\ell_d^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	178	62.8	56.3	55.6
$\mathbf{t}^{[1]}$	169	59.6	53.4	52.8
$\mathbf{t}^{[11]}$	179	63.3	56.7	56.1
$\mathbf{t}^{[21]}$	183	64.7	57.9	57.3

λ_3

$\ell_p^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	162	56.1	52.7	52.1
$\mathbf{t}^{[1]}$	155	53.5	50.3	49.7
$\mathbf{t}^{[11]}$	174	60.1	56.5	55.8
$\mathbf{t}^{[21]}$	178	61.5	57.8	57.1

λ_3

$\ell_d^{[j,k]}$	$\mathbf{s}^{[0]}$	$\mathbf{s}^{[1]}$	$\mathbf{s}^{[11]}$	$\mathbf{s}^{[21]}$
$\mathbf{t}^{[0]}$	186	64.4	60.5	59.8
$\mathbf{t}^{[1]}$	175	60.6	56.9	56.3
$\mathbf{t}^{[11]}$	189	65.3	61.3	60.6
$\mathbf{t}^{[21]}$	192	66.3	62.3	61.5

Table S2: *The effect of different choices of coarse-graining parameters* $[j,k]$ *for tangents and arc lengths on the value of* $\ell_p^{[j,k]}$ *and* $\ell_d^{[j,k]}$ *for four different sequences of length* 300 *bp: poly(AA), poly(AT), poly(GG) and* λ_3

AA		AG		CG		GG		TA		TG	
MC	MD	MC	MD	MC	MD	MC	MD	MC	MD	MC	MD
0	0	0	0	0	0	0	0	0	0	0	0
-0.75	-0.82	-0.78	-0.98	-0.72	-1.01	-1.10	-1.26	-0.55	-0.67	-0.65	-0.81
-1.51	-1.48	-1.89	-2.30	-2.29	-1.90	-2.77	-3.20	-2.65	-2.78	-2.62	-2.64
-2.28	-2.07	-2.68	-3.76	-2.56	-1.96	-4.78	-5.56	-2.91	-3.22	-3.23	-3.15
-2.96	-2.55	-4.07	-5.41	-5.17	-4.15	-6.72	-7.90	-6.20	-6.15	-6.26	-5.94
-3.51	-2.99	-4.59	-6.46	-5.48	-4.14	-8.15	-9.64	-6.47	-6.60	-7.18	-6.55
-3.87	-3.38	-5.28	-6.96	-6.81	-5.65	-8.78	-10.37	-8.49	-8.13	-8.35	-7.92
-4.08	-3.69	-5.41	-7.11	-7.41	-6.17	-8.60	-10.01	-8.88	-8.50	-9.40	-8.43
-4.23	-4.10	-5.24	-6.39	-6.66	-5.67	-7.83	-8.79	-8.41	-7.93	-7.80	-7.51
-4.44	-4.49	-5.42	-6.03	-7.44	-6.81	-6.93	-7.24	-8.93	-8.29	-8.63	-7.82
-4.80	-4.93	-5.37	-5.42	-6.37	-5.36	-6.34	-5.88	-7.59	-7.31	-6.61	-6.50
-5.33	-5.36	-5.95	-5.65	-6.99	-6.31	-6.43	-5.18	-8.05	-7.61	-7.21	-6.89

Table S3: Expectations $\ln\langle \mathbf{t}_i \cdot \mathbf{t}_0 \rangle$ evaluated on MC and MD ensembles for 18bp fragments containing the six distinct dimer steps. $\mathbf{t}_0$ is the basepair normal to the fourth base pair from one end, and the row index $i = 1, \ldots, 11$ runs until the fourth basepair from the other end. For formatting reasons, actual values are table entries divided by 100.

AA		AG		CG		GG		TA		TG	
MC	MD	MC	MD	MC	MD	MC	MD	MC	MD	MC	MD
0	0	0	0	0	0	0	0	0	0	0	0
-0.11	-0.11	-0.06	-0.17	-0.02	-0.19	-0.40	-0.54	-0.00	-0.00	-0.01	-0.01
-0.38	-0.28	-0.70	-1.06	-0.93	-0.63	-1.50	-2.02	-1.13	-1.15	-1.27	-1.07
-0.68	-0.40	-0.92	-1.96	-0.70	-0.20	-2.94	-3.94	-1.02	-1.18	-1.43	-1.19
-0.90	-0.41	-1.85	-3.16	-2.60	-1.90	-4.27	-5.81	-3.30	-3.09	-3.73	-3.18
-0.96	-0.37	-1.76	-3.68	-2.39	-1.39	-5.07	-7.08	-3.19	-3.14	-4.17	-3.46
-0.86	-0.27	-1.98	-3.74	-2.95	-2.41	-5.08	-7.34	-4.09	-3.65	-4.54	-4.06
-0.62	-0.17	-1.52	-3.35	-3.03	-2.40	-4.29	-6.55	-4.11	-3.64	-5.09	-4.26
-0.34	-0.09	-0.91	-2.21	-1.55	-1.44	-2.97	-4.90	-2.61	-2.17	-2.77	-2.52
-0.11	-0.08	-0.51	-1.36	-1.82	-2.00	-1.53	-2.96	-2.74	-2.14	-3.13	-2.50
-0.02	-0.13	-0.03	-0.30	-0.13	-0.21	-0.43	-1.24	-0.50	-0.30	-0.49	-0.48
-0.11	-0.22	-0.05	-0.04	-0.25	-0.66	-0.00	-0.19	-0.58	-0.23	-0.63	-0.50

Table S4: Data $\ln \hat{\mathbf{t}}_i \cdot \hat{\mathbf{t}}_0$ analogous to Table S3, but now evaluated on MC and MD ground-state shapes. Again actual values are table entries divided by 100.

References

[1] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 1996.

[2] George Marsaglia. Xorshift RNGs. *J Stat Softw*, 8(14):1–6, 2003.

[3] George Marsaglia and Wai Wan Tsang. The Ziggurat Method for Generating Random Variables. *J Stat Softw*, 5(8):1–7, 2000.

[4] E. Anderson, Z. Bai, C. Bischof, L. S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. Society for Industrial and Applied Mathematics, Philadelphia, third edit edition, January 1999.

[5] Luke Czapla, David Swigon, and Wilma K. Olson. Sequence-dependent effects in the cyclization of short DNA. *J Chem Theory Comput*, 2(3):685–695, 2006.

[6] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.*, 21(6):1087–1092, 1953.

[7] Filip Lankas, Oscar Gonzalez, L M Heffler, G Stoll, M Moakher, and John H Maddocks. On the parameterization of rigid base and basepair models of DNA from molecular dynamics simulations. *Phys. Chem. Chem. Phys.*, 11(45):10565–10588, December 2009.

[8] F. Sanger, A. R. Coulson, G. F. Hong, D. F. Hill, and G. B. Petersen. Nucleotide sequence of bacteriophage λ DNA. *J Mol Biol*, 162(4):729–773, 1982.

[9] Johanna Virstedt, Torunn Berge, Robert M. Henderson, Michael J. Waring, and Andrew A. Travers. The influence of DNA stiffness upon nucleosome formation. *J. Struct. Biol.*, 148(1):66–85, 2004.

[10] J Bednar, P Furrer, V Katritch, A Z Stasiak, J Dubochet, and A Stasiak. Determination of DNA persistence length by cryo-electron microscopy. Separation of the static and dynamic contributions to the apparent persistence length of DNA. *J. Mol. Biol.*, 254(4):579–594, 1995.

[11] J D Kahn and D M Crothers. Protein-induced bending and DNA cyclization. *Proc. Natl. Acad. Sci. USA*, 89(14):6343–6347, 1992.

[12] Stephanie Geggier and Alexander Vologodskii. Sequence dependence of DNA bending rigidity. *Proc. Natl. Acad. Sci. USA*, 107(35):15421–15426, 2010.