

A Practical Guide to Molecular Dynamics Simulations of DNA Origami Systems

Jejoong Yoo Chen-Yu Li Scott Michael Slone Christopher Maffeo

Aleksei Aksimentiev

August 11, 2017

Abstract

The DNA origami method exploits the self-assembly property of nucleic acids to build diverse nanoscale systems. The all-atom molecular dynamics (MD) method has emerged as a powerful computational tool for atomic-resolution characterization of the *in situ* structure and physical properties of DNA origami objects. This chapter provides step-by-step instructions for building atomic-scale models of DNA origami systems, using the MD method to simulate the models, and performing basic analyses of the resulting MD trajectories.

How to cite this work: Yoo J., Li CY., Slone S.M., Maffeo C., Aksimentiev A. (2018) A Practical Guide to Molecular Dynamics Simulations of DNA Origami Systems. In: Zuccheri G. (eds) DNA Nanotechnology. Methods in Molecular Biology, vol 1811. Humana Press, New York, NY DOI:10.1007/978-1-4939-8582-1_15

1 Introduction

The DNA origami method utilizes the strong specificity of complementary DNA molecules to enable low-cost fabrication of custom three-dimensional nanoscale objects (*1, 2*). In the DNA origami method, distant parts of a long DNA strand are brought into close spatial proximity by association with a collection of short (~20–60 bp) synthetic DNA strands (*3, 4*). The resulting structures appear to be made of many approximately parallel DNA duplexes connected by a network of four-arm Holliday junctions. Although the method was first described only ten years ago, practical DNA origami devices have already been demonstrated (*5–10*), which includes systems for performing nanoscopic measurements (*11–13*), focusing electromagnetic fields below the diffraction limit (*14, 15*), and enhancing enzymatic catalysis (*16, 17*).

The rise of the DNA origami method may in part be attributed to the method's abstract representation of complex self-assembled objects. Several interactive software packages, including Tiamat (*18*),

Nanoengineer-1, and caDNAno (19), allow the user to intuitively draw the parallel helices in desired locations while automated algorithms perform the more tedious tasks, such as sequence assignment for the short synthetic “staple” strands. Of these software packages, caDNAno has emerged as the most widely used, due in part to its easy-to-use interface, its ability to automatically generate staple strands, and its ability to scale to large structures.

The DNA origami design software packages model origami objects as collections of idealized, rigid rods, each rod representing a DNA duplex. However, recent high-resolution cryo-electron microscopy reconstruction of a DNA origami object revealed considerable deviations from the idealized design (20). The time and cost associated with experimental characterization of DNA origami structure make structure prediction through computational approaches an attractive alternative.

Simulations of DNA origami objects have been performed using models that provide different levels of detail including both coarse-grained (21–25) and atomistic models (26–31). In general, coarse-grained descriptions trade accuracy for lower computational costs. Since atomic features are often important at the nanometer scale, the focus of this chapter is on atomistic simulations. In particular we describe the use of all-atom molecular dynamics (MD) simulations, wherein the motion of every atom in the system is described according to classical mechanics. Our group has used this method previously to characterize and predict the structural, mechanical, chemical, and transport properties of DNA origami objects (26–31).

In this chapter, we describe the protocol for all-atom MD simulations of DNA origami using the NAMD (32) package starting from a caDNAno design. As an example, we use a caDNAno design file of a six-helix cylindrical structure (**hextube.json**), which is provided along with this document. We will convert the caDNAno file to an all-atom structure file in the PDB format, submerge the structure in MgCl₂ solution, and perform MD simulations of the solvated structure.

All files necessary to perform modeling tasks described in this chapter can be found at <http://bionano.physics.illinois.edu/sites/default/files/origamitutorial.tar.gz>. It is recommended that you follow this chapter’s instructions on a Linux workstation. If you are a windows user, please change all the file path separators “/” into “\”. Please contact the Aksimentiev Group for any questions about this chapter.

All URLs, file and directory names, and commands that will be manually entered are marked in **bold**. When typing the commands, please enter them verbatim unless directed otherwise. Note that we will type commands in two different terminals. One is the UNIX/LINUX terminal that runs a shell process (e.g., BASH, CSH, TCSH etc.). “SHELL> **COMMAND**” indicates that **COMMAND** should be typed in the

shell terminal. The other is the VMD Tk console terminal; in the VMD menu, select Extensions and Tk Console. “TKCON> **COMMAND**” indicates that **COMMAND** should be typed in the VMD Tk console.

2 Materials

2.1 Software

- 1. caDNAno.** Download caDNAno 2 from <http://cadnano.org>. caDNAno is one of the most popular computer-aided origami design programs. caDNAno enables rapid design of DNA origami objects, runs on Windows and Mac platforms, and can be run as a plugin of Autodesk Maya or as a standalone program. For more information on caDNAno, please refer to the caDNAno manual.
- 2. ENRG MD web server.** Access it at <http://bionano.physics.illinois.edu/origami-structure>. The ENRG MD web server converts caDNAno output files to an appropriate format for visualization and simulation. As the tool is available entirely online, it requires no installation. It can convert DNA origami structures using the square or honeycomb lattices, and can use the default m13mp18 template or a custom user-provided sequence for the DNA scaffold.
- 3. VMD.** Download VMD from <http://www.ks.uiuc.edu/Research/vmd>. VMD is a program for visualization and analysis of atomic biomolecular structures (33). VMD supports all major computer platforms and provides a programming interface based on the Tcl scripting language. Tcl scripts will be used throughout this chapter for both building and analyzing DNA origami objects. VMD runs on any system, but benefits greatly from the latest graphics hardware that supports CUDA. For more information on VMD, please refer to the VMD User Guide (34) and the VMD Tutorial (35). After installation, VMD is run either by typing **vmd** in the terminal (Linux systems) or by clicking the icon (Windows and Mac systems).
- 4. NAMD.** Download NAMD (32) from www.ks.uiuc.edu/Research/namd. NAMD is a highly scalable molecular dynamics code that supports CUDA-based acceleration. NAMD can be used on any machine (from laptop to supercomputers) running Linux/UNIX, Mac OS X, or Windows operating systems. However, for DNA origami systems, it is highly recommended to perform simulations using multiple compute nodes on parallel clusters or supercomputers. For more information on NAMD, please refer to the NAMD User Guide (36) and NAMD Tutorial (37).
- 5. Perl.** Download Perl from <https://www.perl.org/get.html>. Perl is a general-purpose scripting language available across all major platforms. It is used in this chapter to generate magnesium hexahydrate restraints and to convert an atomic model of a DNA origami object into a sparser “chickenwire” model for the purpose of visualization.

2.2 Required files

1. origamiTutorial package. All files used in this chapter (scripts and other support files) are available in an archive on the publisher website or at <http://bionano.physics.illinois.edu/sites/default/files/origamitutorial.tar.gz>. Extracting the archive in a working directory on your filesystem will create the **origamiTutorial** directory containing four subdirectories. Each subdirectory contains files required by one section in this chapter. The subdirectories extracted from **origamitutorial.tar.gz** are labeled **step1**, **step2**, **step3**, and **step4**, corresponding to sections §3.1, §3.2, §3.3, and §3.4 of this chapter.

2. CHARMM force field with corrections for ion–DNA interactions. Download the topology and parameter files of the CHARMM force field from http://mackerell.umaryland.edu/charmm_ff.shtml. Select the c36 version. Download the CHARMM stream file for water and ions with corrections for ion–DNA interactions from <http://bionano.physics.illinois.edu/CUFIX>. The CHARMM topology file for nucleic acids, **top_all36_na.rtf**, the CHARMM parameter file for nucleic acids, **par_all36_na.prm**, and the stream file for water and ions **toppar_water_ions_cufix.str** are used in this tutorial. These three files are also included in the **origamitutorial.tar.gz** package.

3 Methods

Here we describe our protocols for preparing models of DNA origami systems and simulating these models using the MD method. If you wish to design a DNA origami system, go to section §3.1. The three following sections—§3.2, §3.3, and §3.4—cover construction, simulation, and analysis of atomistic models of a DNA origami system, respectively. In section §3.2, we describe methods for converting an abstract design of a DNA origami system obtained in section §3.1 to an atomistic model suitable for MD simulations. In section §3.3, we explain the strategy for equilibrating the atomistic DNA origami structure in an electrolyte solution. In section §3.4, we introduce a few general analysis methods that are useful for monitoring the equilibration process.

3.1 Designing a DNA origami object using caDNAno

Our first task is to build a DNA origami object using the caDNAno program and the ENRG MD web server. An MD simulation requires two separate files, a PDB file and a PSF file, to define a system. The PDB file contains coordinates of all atoms in the system. The PSF file is complementary to the PDB file and contains chemical information about the system including masses and charges of the atoms as well as

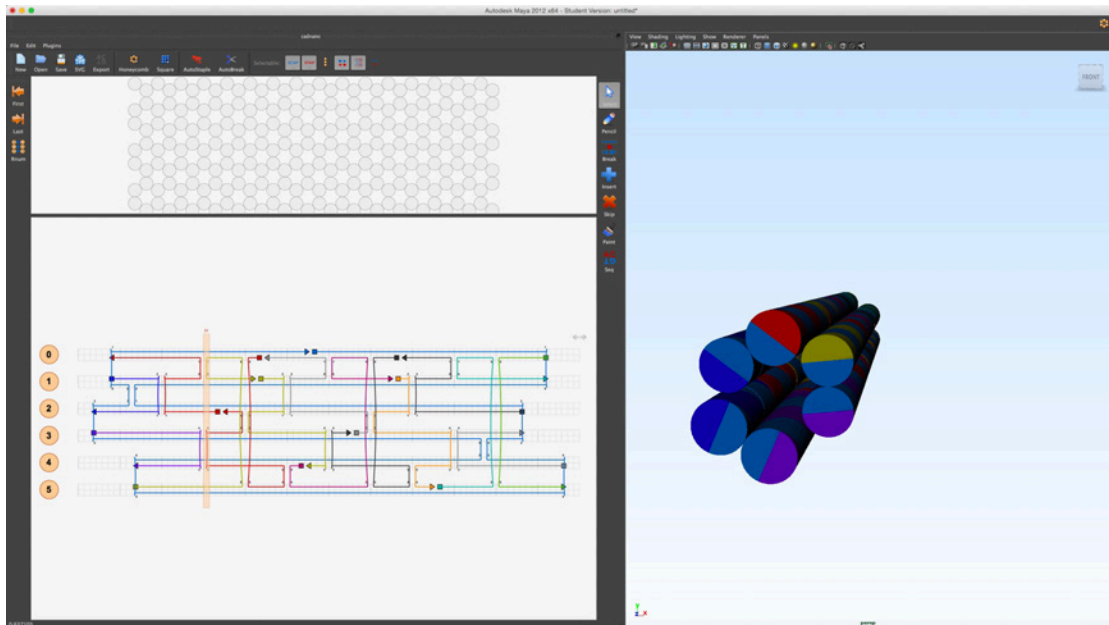


Figure 1: A screenshot of a caDNAno session showing the design of our DNA origami example: hextube.

chemical bond information. Here we construct a hexagonal tube structure that consists of six DNA helices. The scripts used in this section are located in the **step1** subdirectory.

1. **Open caDNAno.** First open caDNAno by navigating to either the **cadnano2** or **cadnano2.5** directory.

cadnano2: SHELL> python main.py

cadnano2.5: SHELL> cadnano

2. **Start a new session.** Click “New” and choose a honeycomb type. Fig. 1 shows an example session of caDNAno.
3. **Choose six helices in a hexagonal arrangement,** as shown in Fig. 1.
4. **Draw the DNA paths to produce the hexagonal tube structure.**
5. **Save the design.** Click “Save”. The resulting file is **hextube.json**, which can be found in the **step1** folder. One can follow the remaining steps using any custom-design file instead of **hextube.json**.

3.2 Building an optimized atomistic structure

Next, we convert the **hextube.json** file to a structure file in the PDB format using the server, <http://bionano.physics.illinois.edu/origami-structure>, as shown in Fig. 2.

Name your job (optional). *

Upload a DNA origami design .json file

hextube.json

Select the origami lattice. *

Square

Honeycomb

Select the scaffold sequence. *

m13mp18 (up to 7,249 bases)

Custom

Download your [archive](#). Your files will be deleted from our server in 2 hours.

Figure 2: Using the ENRG MD web server to convert a caDNAno design file to a PDB-formatted atomic structure. Make sure your selections match exactly what is shown.

As shown in Fig. 2, we named our project “**hextube**” and uploaded the **hextube.json** file. We set the origami lattice type to honeycomb. For the scaffold, one can select the default m13mp18 sequence or choose to upload a custom DNA sequence.

1. **Open a web browser.** Any web browser will work.
2. **Open the ENRG MD web site.** With any web browser, navigate to <http://bionano.physics.illinois.edu/origami-structure>. An example of the ENRG MD web server session is shown in Fig. 2.
3. **Name the session.** Type in the job title hextube.
4. **Upload json file.** Click “Choose file” and select **hextube.json**.
5. **Select the origami lattice type.** The hexagonal tube structure is designed based on the honeycomb lattice, so choose “Honeycomb”.
6. **Select the scaffold sequence.** Choose the default M13 vector sequence.
7. **Create simulation files.** Clicking “Create simulation files” will initiate the conversion process, which will generate a download link upon completion.

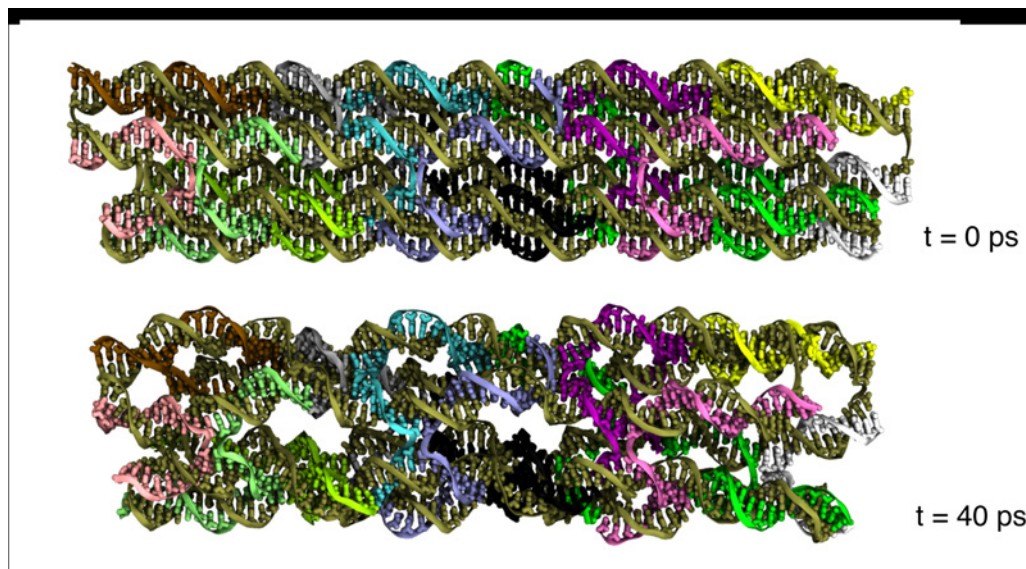


Figure 3: The hexagonal tube structure before (top) and after (bottom) structure optimization. A 40 ps ENRG MD simulation performed in vacuum with the extrabonds restraints is sufficient to produce the expected chickenwire arrangement of the DNA helices. The molecular graphics images were obtained using VMD's , Drawing Methods "Licorice" for the "not backbone and noh" selection and "NewCartoon" for the "backbone" selection; the Coloring Method was set to "SegName".

8. Download the simulation files. Download the simulation files by clicking the download link. The file name of the downloaded archive will be **hextube.tar.gz**. Please put the hextube.tar.gz file in the **origamiTutorialNamd** folder and extract it. The extracted **hextube** directory will contain the following files:

- hextube.psf: The CHARMM format PSF file.
- hextube.pdb: The PDB format file of the all-atom structure.
- hextube.namd: The NAMD input file for the structure optimization simulations.
- hextube.exb: Extrabonds restraints for structure optimization.
- charmm36.nbf: CHARMM36 force field files.

If a custom JSON file was created in §3.1, one can remove the **step2** directory and change **hextube** directory to **step2**. Otherwise, one can use the existing **step2** directory, which includes all files from **hextube.tar.gz**.

9. Check the downloaded file. Now, one can see the three-dimensional structure of the DNA origami object by loading the PSF and PDB files to the VMD program. In VMD Main window, click "File" and "New Molecule", navigate to **hextube.psf** file in **step2** directory using the "Browse" button, and click "Load" button. After loading the PSF file, a molecule with "ID 0" will appear in "VMD Main" window. Click the newly created line in "VMD Main" window,

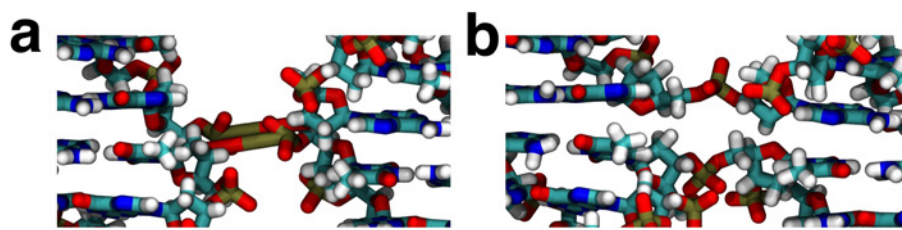


Figure 4: Holliday junction conformations before (a) and after (b) the optimization using the NAMD. The molecular graphics images were obtained using VMD's Drawing Method "Licorice" and Coloring Method "Name".

click "File" and "Load Data Into Molecule", navigate to **hextube.pdb** file, and click "Load" button. Alternatively, a new molecule can be loaded to VMD by typing the following command to the VMD Tk console: "TKCON> **mol load psf hextube.psf pdb hextube.pdb**". Then, a molecular structure will appear in the "VMD OpenGL Display" window. For example, Fig. 3 (top) shows the structure of **hextube.pdb** obtained using the web server. The structure in Fig. 3 (top) is unrealistic because the DNA helices are parallel to one another and the bonds in the Holliday junctions are abnormally stretched, Fig. 4a.

As described in our previous study (31), a more realistic structure can be obtained by performing an ENRG MD simulation of the structure in vacuum using the network of elastic restraints (**hextube.exb**) provided by the web server. See the NAMD manual for a detailed description of the extrabonds command. (e.g., <http://www.ks.uiuc.edu/Research/namd/2.7/ug/node26.html>).

10. Prepare a NAMD run. The initial optimization of the structure can be performed by running an MD simulation using the NAMD input file, **hextube.namd**, provided by the web server. Please modify the restart frequency ("restartfreq") and number of simulation steps ("run") as such:

```
restartfreq      4800
run              21600
```

11. Note the use of extrabonds. Extrabonds serve two different purposes: to imitate the DNA–DNA repulsion observed in explicit solvent simulations and to enforce stability of the DNA helices. The extrabonds are specified in the NAMD input file as follows:

```
extraBonds on
extraBondsFile hextube.exb
```

12. Run NAMD. Perform the MD simulation by typing the following into a terminal:

```
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2
+p14_hextube.namd > hextube.log &
```


13. Monitor the NAMD run. The MD run will write output files into the output directory. In practice, the required simulation time depends on the size of the DNA origami object. For a relatively small system such as the hexagonal tube object, 40 ps seems long enough to get a realistic conformation, Fig. 3 (bottom). For a ~7500-bp object, 2 ns was sufficient to achieve a fully relaxed structure (31). If you are using a multicore system you can improve simulation speed by using the +p option specifying the number of cores (e.g. /home/ubuntu/NAMD Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14 hextube.namd > hextube.log &).

14. Save the optimized structure to PDB. Before we proceed, we need to save the last frame of the trajectory as a PDB file for the tasks described in the next section:

```
SHELL> vmd -dispdev text -e ../step3/save_pdb.tcl -args hextube.psf \  
output/hextube-1.coor hextube_min.pdb
```

Note that the backslash above indicates that the command should be a single line. An example of this output (**hextube_min.pdb**) has been placed in the **step2** directory, along with the initial PSF and PDB files.

3.3 MD simulations of DNA origami in explicit MgCl₂ solution

As each nucleotide of DNA carries the charge of an electron, a DNA origami object has a high charge density. In experiment, DNA origami objects are usually stabilized by divalent cations, typically Mg²⁺. In this section, we describe a protocol for simulating a DNA origami structure in an explicit electrolyte solution. All files necessary for the simulations are included in the **step3** folder. Please navigate to the **step3** folder before proceeding to the following steps.

1. **Insert Mg²⁺.** In this step, we neutralize a DNA origami object by inserting the appropriate number of Mg²⁺ ions. Previously, we have shown that the standard parameterization of Mg²⁺ ions is not suitable for simulations of dense DNA systems (38). We have also shown that modeling a Mg²⁺ ion as a magnesium hexahydrate complex (MGHH²⁺) brings simulated DNA–DNA forces in agreement with experiment (38, 39,40). Hence, all our DNA origami simulations employ a MGHH²⁺ model of Mg²⁺ ions (26). Because MGHH²⁺ molecules diffuse slowly, it is beneficial to place them initially in proximity to the DNA origami structure. The following custom script will place MGHH²⁺ molecules close to the DNA origami. In the terminal, execute:

```
SHELL> vmd -dispdev text -e add_mgh_ver2_3.tcl -args ../step2/hextube.psf \  
../step2/hextube_min.pdb hextube_MGHH 516
```

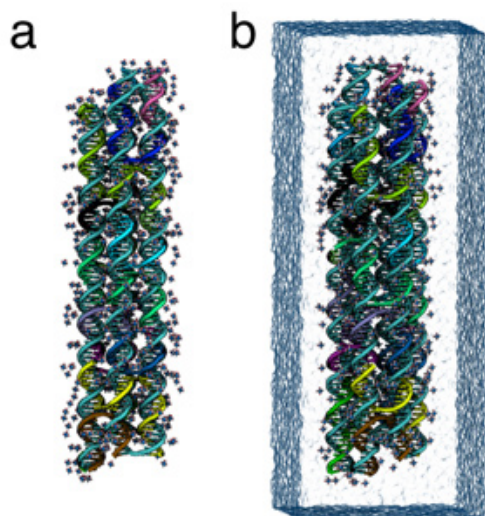


Figure 4: Hexagonal tube structure including MGHH^{2+} (a) and the solvation box (b). The molecular graphics images were obtained using VMD's Drawing Method "New Cartoon" with "Aspect Ratio" 1.0 and "Thickness" 1.5 for the selection "backbone" and Drawing Method "Licorice" with "Bond Radius" 0.8 for the selection "nucleic and not backbone and noh". Coloring Method for DNA was "SegName". The transparent water box was drawn using Drawing Method "MSMS" with "Probe Radius" 5.5 and Material "GlassBubble" for the selection "name OH2".

t

The first two arguments **step2/hextube.psf** and **step2/hextube_min.pdb** are the PSF and PBD files from the last frame of the structure optimization simulation. The third argument "hextube_MGHH" is the prefix for the output PSF and PDB files. The last argument "516" is the number of MGHH^{2+} molecules to be added, which can be adjusted depending on the size of the final simulation box and the desired concentration of Mg^{2+} . The resulting system should look like the one shown in Fig. 5a.

15. Solvate the system. In experiment, DNA origami objects self-assemble in aqueous solutions. In order to simulate a DNA origami object under experimental conditions, we need to surround the object with solution. We use the **sollon.tcl** script to add water to the system. The script also adds Cl^- ions to make the entire system electrically neutral. In the terminal, run the following command:

```
SHELL> vmd -dispdev text -e sollon.tcl
```

The above command will produce the final solvated hexagonal tube system (**hextube_MGHH_WI.psf** and **hextube_MGHH_WI.pdb**). Note that the number of monovalent ions can be adjusted by modifying **sollon.tcl**. Fig. 5b shows an image of the solvated structure.

16. Create extrabonds for MGHH^{2+} . The six water molecules in an MGHH^{2+} complex are strongly coordinated through direct contact interactions with the Mg^{2+} ion. However, the structure of MGHH^{2+} can be easily broken during an MD simulation of a DNA origami object because of the large electrostatic forces between DNA and Mg^{2+} . To prevent undesired dissociation of the MGHH^{2+} complexes, we add harmonic restraints between the magnesium atom and the oxygen atoms of the water in each MGHH^{2+} using the extrabonds function of NAMD. Running the **mggh_extrabonds.pl** script in the terminal will generate extrabonds definitions for NAMD.

```
SHELL> perl mghh_extrabonds.pl hextube_MGHH_WI.pdb >
mghh_extrabonds
```

Alternatively, you can use the Tcl version of the same script:

```
SHELL> tclsh mghh_extrabonds.tcl hextube_MGHH_WI.pdb >
mghh_extrabonds
```

To use the extrabonds restraints in an NAMD simulation, the following lines need to be added to the NAMD configuration script:

```
extraBonds on
extraBondsFile ./mghh_extrabonds
```

These lines are already included in all NAMD scripts used in this tutorial. If you plan to use extrabonds restraints in a future simulation, remember to include the files in your version of the NAMD configuration script. All PSF, PDB, and extrabonds files produced in this chapter are in the **step3** directory.

17. Equilibrate the solvated structure using NAMD. Equilibrating a DNA origami structure is not a trivial task. There are two main forces that could break the local structure of a DNA origami object. One of them is the electrostatic repulsion between the phosphates of the DNA strands. The other is the torsion at the crossovers. To avoid breaking base pairs while relaxing the global structure, we simulate a DNA origami object with an elastic network model (ENM) of harmonic restraints that reinforce base pairing and base stacking. We gradually reduce the strength of the spring constants before running a fully unrestrained equilibration simulation. The following protocol will simulate the hexagonal tube with ENM restraints using a force constant $k = 0.5, 0.1, 0.01$ and 0 (kcal/mol/Å²) for 4.8 ns for each value of k . Although this protocol works for most small-to-medium DNA origami structures, one may need to modify the protocol, in particular the duration of a free equilibration simulation ($k=0$), to achieve a proper relaxation of a DNA origami structure. The following command run in a terminal window will generate ENM extrabonds files with the three force constants:

```
SHELL> sh mk_extra.sh
```

For the $k = 0$ simulation, we simply turn off the ENM constraints.

Then, run the simulations one after another:

```
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14
equil_min.namd > equil_min.log
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14
equil_k0.5.namd > equil_k0.5.log
```

```
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14
      equil_k0.1.namd > equil_k0.1.log
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14
      equil_k0.01.namd > equil_k0.01.log
SHELL> /home/ubuntu/NAMD_Git-2018-06-26_Linux-x86_64-multicore/namd2 +p14
      equil_k0.namd > equil_k0.log
```

Note that each run except the first run is a continuation from the previous run using the restart files; for example, `equil_k0.1.namd` input file uses three restart files created by `equil_k0.5.namd`, `equil_k0.5.restart.coor`, `equil_k0.5.restart.vel`, and `equil_k0.5.restart.xsc`. Each run generates a trajectory file (DCD) and box size file (XST). Example DCD and XST output files can be found in the `step4` directory for the readers who want to proceed without actually running the simulations. Keep in mind these are significantly compressed versions of what you will produce if you don't modify the NAMD scripts. If your simulations in the `step3` directory are successful, copy newly generated DCD, XST, PSF, and PDB files to the `step4` directory:

```
SHELL> cp *.dcd *.xst *.psf *.pdb ../step4
```

It is strongly recommended to monitor your simulations by reading the simulation logs, see Note 1. If your simulation does not run or it ends sooner than expected, the simulation log might specify the reason; see Notes 2-4 for an explanation of different error types.

3.4 Analysis of the MD trajectories

In this section, we will analyze the resulting simulation trajectories using the provided scripts. Specifically, we will first determine the change of the simulation box size during the simulation. Then we will calculate the Root Mean Square Deviation (RMSD) of the hexagonal tube with respect to its initial coordinates. We will next measure the total charge surrounding the hexagonal tube. Finally, we will count the number of broken base pairs. Please note that there are other ways to assess whether a simulation has reached equilibrium (26, 27, 41).

The script files for the analysis are included in `step4` folder. Please make sure you are in the `step4` folder before proceeding.

- 1. Size of the simulation box.** In the previous section, we used the “solvate” plugin of VMD to add water to the simulation system, which usually underestimates the number of water molecules needed to fill the simulation box containing a DNA origami object. As a result, the simulation box shrinks when the system is simulated under 1 atm of pressure. Thus, the change of the simulation box size can be used to monitor the process of equilibration. The following command

will extract the box size information from the **equil_kN.xst** file ($N = 0.5, 0.1, 0.01$ and 0) and print the box size along the x , y , and z axes in the **equil_kN_X.dat**, **equil_kN_Y.dat** and **equil_kN_Z.dat** files:

```
SHELL> sh grepBoxTrace.sh
```

In the output file, the first column is the simulation time in nanoseconds, and the second column is the length of the object in Angstroms. You can use the plotting software of your choice (e.g. xmgrace) to visualize the data. The result should look similar to that shown in Fig. 6. The simulation box should shrink in the first 300 ps. After that, the box size should become stable.

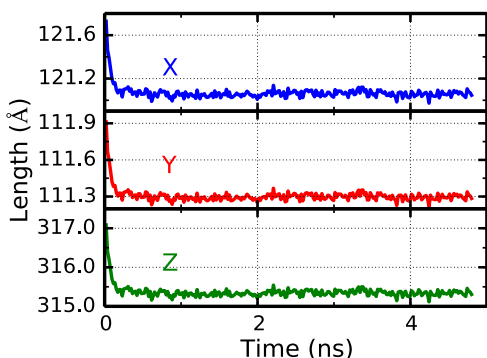


Figure 6: The size of the simulation box versus the simulation time.

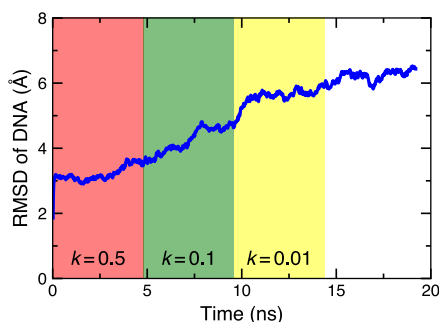


Figure 7: The RMSD of the hexagonal tube DNA origami from its idealized design

2. RMSD. In this step, we will measure RMSD, which characterizes the amount by which a given selection of atoms deviates from its initial coordinates. The script **measureRMSD.sh** will take all the trajectories, measure the RMSD of each frame relative to the initial structure in **hextube_MGHH_WI.pdb**, and write the data to **sim_RMSD.dat**.

```
SHELL> sh measureRMSD.sh
```

In the output file (**sim_RMSD.dat**), the first column is the simulation time in nanoseconds, and the second column is the RMSD in Angstroms. If you plot the first column as the x -axis and the second column as the y -axis, the graph should look like the one in Fig. 7. The hexagonal tube is a relatively simple DNA origami structure. After the first 15 ns, the RMSD reached a nearly constant value.

3. Counting the number of broken base pairs. Computing RMSD is a method for monitoring the integrity of the simulated structure. Counting the number of broken base pairs is another effective method specific to DNA origami structures. The script, **countBrokenBps.tcl** achieves

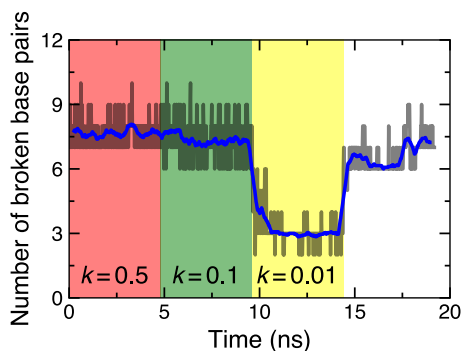


Figure 8: The number of broken base pairs in the hexagonal tube DNA origami structure.

this by computing the number of intact base pairs for each frame of an MD trajectory and subtracting that number from the number of base pairs in the idealized structure. In the script, a base pair is considered intact if the H1 or N1 atom of a purine is within 3 Å of the N3 or H3 atom of a pyrimidine, and the angle formed by the N1–H1–N3 or N1–H3–N3 atoms is greater than 140 degrees. These atoms participate in the central Watson–Crick hydrogen bond. Type the following command in the terminal to count the number of broken base pairs in the DNA origami and save the output to `sim_BrokenBps.dat`.

```
SHELL> sh countBrokenBps.sh
```

In the output file (`sim_BrokenBps.dat`), the first column is the simulation time in nanoseconds, and the second column is the number of broken base pairs. The gray trace is the raw data. The blue trace is the running average obtained using a 0.5 ns window.

4. Total charge. To neutralize the DNA origami charge, we added MGHH^{2+} molecules. When the system reaches equilibrium, the ionic atmosphere around the DNA origami object should also reach equilibrium. To characterize equilibration of the ion atmosphere, we measure the total charge of all atoms residing within 2 nm of the DNA origami structure as a function of the simulation time. Because we did not invoke the “wrapAll” option in the NAMD configuration file, atoms could wander from one periodic image of the system to the other and hence we need to wrap the trajectories first. The following command will call `AlignWrap.tcl`, to place all atoms of the MD trajectories into the same periodic image of the system and save the output.

```
SHELL> sh AlignWrap.sh
```

In VMD, you can load the DCD file before wrapping and after wrapping to compare the difference. Now, the wrapped DCD files are ready for analysis. Type the following command in the terminal to measure the charge of the DNA origami and its environment and save the output to `sim_Charge.dat`.

```
SHELL> sh measureCharge.sh
```

In the output file (**sim_Charge.dat**), the first column is the simulation time in nanoseconds, and the second column is the charge due to all atoms within 2 nm of the DNA atoms in the units of the elementary charge, the charge of a proton. If you plot the first column as the x -axis and the second column as the y -axis, the graph should look like the one in Fig. 9. The gray trace is the raw data. The blue trace is the running average obtained using a 0.5 ns window.

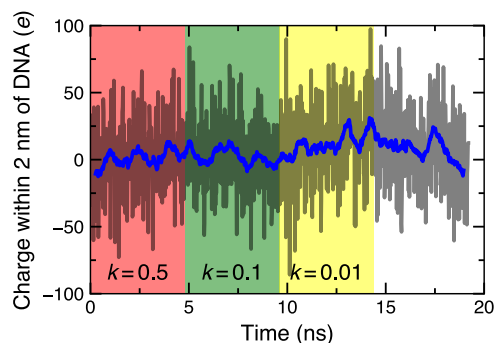


Figure 9: The total charge of the DNA origami object and the surrounding electrolyte.

5. Chickenwire representation. DNA origami objects are complex macromolecules that consist of hundreds to thousands of nucleotides distributed over multiple DNA strands. The atomic representations commonly used to visualize the outcome of an MD simulation are sometimes too detailed, which can make it difficult to appreciate the global conformation of a DNA origami object. An alternative approach is to use the chickenwire representation (20), which can complement the atomic-level representations of the structures.

In the chickenwire representation, a base pair is represented by a single point placed at its center of mass. The chickenwire representation is drawn by connecting such points if the two base pairs are stacked in a helix or linked by a crossover. Fig. 10 illustrates the outcome of each modeling procedure described in sections §3.1, §3.2, and §3.3 using the chickenwire representation.

6. The conversion from an atomistic to a chickenwire representation can be done directly provided you have the additional files in the archive from the ENRG MD server (§3.1). You will also need the last frame of your equilibration simulation. The script **writePDB.tcl** will do this automatically, by loading **equil_k0.dcd** and saving the last frame as a PDB file:

```
SHELL> vmd -dispdev text -e writePDB.tcl
```

Note that conversion to a chickenwire representation applies only to atoms of DNA, which are selected in the script as:

```
[atomselect top nucleic]
```

Once the atomic PDB file `equil_k0.pdb` is written, one can convert it to a PDB file for chickenwire using the Perl script `pdb2chickenwire.pl`:

```
SHELL> perl pdb2chickenwire.pl --ndx=../step2/chickenwire.for.make_ndx \  
equil_k0.pdb > chickenwire.pdb
```

Essentially, `pdb2chickenwire.pl` computes the center of mass of each base pair and writes the center of mass coordinates using a PDB file format. To compute the center of mass, the Perl script needs to know which two bases form a base pair. The list of base pairs is given in `chickenwire.for.make_ndx`, which is included in the archive from the ENRG MD server (§3.2).

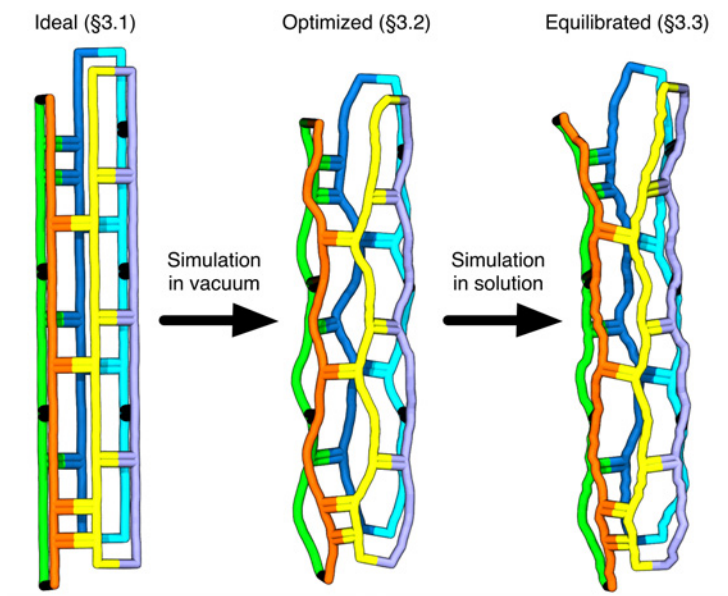


Figure 10: Transformation of the hexagonal tube structure from its ideal design to an equilibrated conformation. Each of the three structures represents the outcome of the modeling tasks described in sections §3.1, §3.2, and §3.3, respectively. To draw the chickenwire structures, Coloring Method “SegName” and Drawing Method “Licorice” with Bond Radius 2.5 were used.

The PDB file produced by the Perl script contains position information required to draw a chickenwire representation but does not contain the connectivity information (base stacking and crossovers). For visualization using VMD, one needs to use the PSF file from the ENRG MD web server archive (§3.2) that contains the necessary connectivity information. One can load the chickenwire PDB file into VMD using the following command in the VMD Tk console:

```
TKCON> mol load psf ../step2/chickenwire.psf pdb chickenwire.pdb
```


4 Notes

1. In the first few steps of equilibration, the system can be unstable. You should check the properties such as the total energy, temperature, and pressure, among others, to make sure there are no abnormal fluctuations. It is also recommended that you read your simulation logfile, which usually has the suffix .log. This file contains information on the above properties, as well as any warnings NAMD might have about your simulation. If your simulation ends unexpectedly, the logfile will also record the exact cause. The causes can vary from simply not finding one of your simulation's input files, to a critical error in the simulation causing it to become unstable. Below are several possible errors you might encounter.
2. Some errors, including Rattle errors and “atoms moving too fast” occur when the atoms in your system are subject to unusually large forces, although they can occasionally occur otherwise. The logfile will identify atoms that are usually very close to the problem area, for example:

ERROR: Constraint failure in RATTLE algorithm for atom 235!

ERROR: Constraint failure; simulation has become unstable.

These errors can occur when the system has not gone through enough steps of energy minimization. If the error occurs right after minimization, ensure that the total energy in the system has stopped decreasing by the end of minimization, otherwise consider minimizing for longer (note, however, that the 4800 steps used in this guide is usually many times longer than necessary). If the problem persists, look closely at the initial structure using VMD, particularly around the affected atoms (for the example above, the selection text “same residue as pbwithin 8 of serial 235” would be useful). It is possible that some atoms may be placed in a configuration that minimization is unable to resolve. For example, problems usually arise when atoms are placed such that a bond pierces an aromatic ring or when two macromolecules are strongly overlapping. Remember to check for clashes with periodic images. If you encounter a problem like this you may need to rebuild the system. If the problem area is small, you may be able to remove problem by selecting and moving atoms using the **moveby** command.

3. Equilibration in NPT allows the box size to fluctuate. This can cause errors if the box size shrinks or expands excessively along any of the system dimensions. This error is usually of the form below.

FATAL ERROR: Periodic cell has become too small for original patch grid!

Possible solutions are to restart from a recent checkpoint, increase margin, or disable useFlexibleCell for liquid simulations.

First, make sure that the dimensions of the system in the configuration file match the physical dimensions of the system. Usually this error only occurs when the option **useFlexibleCell** is turned on. If this option is not turned off, restarting the simulation from a recent checkpoint will likely resolve the issue. It might be necessary to disable Flexible cells and to run the simulation in the NVT (constant volume) ensemble. The NVT ensemble is also recommended if you are running a simulation that includes externally applied forces or electric fields. Running these types of simulations in NPT guarantees this error will eventually occur.

4. It is possible that there can be insufficient water molecules in a simulation system, which will result in vacuum bubbles during equilibration. There are several solutions for this. You can resolvate after equilibration, placing additional water molecules inside the bubbles. You can also run your simulation in NPT with the DNA position fixed, allowing the box size to fluctuate until it reaches a value appropriate for the amount of water molecules in the system. As the latter solution does not involve the placement of additional water molecules, it is the better approach of the two.

Sometimes the bubbles are large enough that their removal would cause an abrupt change in the box size, resulting in unstable MD simulations. In this case, increasing **langevinPistonPeriod** and **langevinPistonDecay** by a factor of 10 can be helpful. By changing these options, the box size is less responsive to the internal pressure, making the MD simulation more stable. After the box size becomes stable, those parameter values should be reverted to their original values.

5 Acknowledgements

This work was supported by the National Science Foundation under Grants DMR-1507985, PHY-1430124, and EEC-1227034.

6 References

1. Seeman NC (2007) An Overview of Structural DNA Nanotechnology. *Mol Biotechnol* 37:246–257. doi: 10.1007/s12033-007-0059-4
2. Pinheiro A V., Han D, Shih WM, Yan H (2011) Challenges and opportunities for structural DNA nanotechnology. *Nat Nanotechnol* 6:763–772. doi: 10.1038/nnano.2011.187
3. Rothmund PWK (2006) Folding DNA to create nanoscale shapes and patterns. *Nature* 440:297–302. doi: 10.1038/nature04586

4. Douglas SM, Dietz H, Liedl T, et al (2009) Self-assembly of DNA into nanoscale three-dimensional shapes. *Science* 325:414–418. doi: <http://dx.doi.org/10.1038/nature08016>
5. Dietz H, Douglas SM, Shih WM (2009) Folding DNA into twisted and curved nanoscale shapes. *Science* 325:725–30. doi: [10.1126/science.1174251](https://doi.org/10.1126/science.1174251)
6. Han D, Pal S, Nangreave J, et al (2011) DNA Origami with Complex Curvatures in Three-Dimensional Space. *Science* 332:342–346. doi: [10.1126/science.1202998](https://doi.org/10.1126/science.1202998)
7. Zadegan RM, Jepsen MDE, Thomsen KE, et al (2012) Construction of a 4 Zeptoliters switchable 3D DNA box origami. *ACS Nano* 6:10050–10053. doi: [10.1021/nn303767b](https://doi.org/10.1021/nn303767b)
8. Andersen ES, Dong M, Nielsen MM, et al (2009) Self-assembly of a nanoscale DNA box with a controllable lid. *Nature* 459:73–76. doi: [10.1038/nature07971](https://doi.org/10.1038/nature07971)
9. Liedl T, Högberg B, Tytell J, et al (2010) Self-assembly of three-dimensional prestressed tensegrity structures from DNA. *Nat Nanotechnol* 5:520–524. doi: [10.1038/nnano.2010.107](https://doi.org/10.1038/nnano.2010.107)
10. Langecker M, Arnaut V, Martin TG, et al (2012) Synthetic lipid membrane channels formed by designed DNA nanostructures. *Science* 338:932–6. doi: [10.1126/science.1225624](https://doi.org/10.1126/science.1225624)
11. Liu S, Su W, Li Z, Ding X (2015) Electrochemical detection of lung cancer specific microRNAs using 3D DNA origami nanostructures. *Biosens Bioelectron* 71:57–61. doi: [10.1016/j.bios.2015.04.006](https://doi.org/10.1016/j.bios.2015.04.006)
12. Nickels PC, Wunsch B, Holzmeister P, et al (2016) Molecular force spectroscopy with a DNA origami-based nanoscopic force clamp. *Science* 354:305–307. doi: [10.1126/science.aah5974](https://doi.org/10.1126/science.aah5974)
13. Schmied JJ, Forthmann C, Pibiri E, Lalkens B (2013) Supporting Information for DNA Origami Nanopillars as Standards for Three- Dimensional Superresolution Microscopy. *Nano Lett* 13:781–785. doi: <http://doi.doi.org/10.1021/nl304492y>
14. Thacker V V., Herrmann LO, Sigle DO, et al (2014) DNA origami based assembly of gold nanoparticle dimers for surface-enhanced Raman scattering. *Nat Commun*. doi: [10.1038/ncomms4448](https://doi.org/10.1038/ncomms4448)
15. Kuzyk A, Yang Y, Duan X, et al (2016) A light-driven three-dimensional plasmonic nanosystem that translates molecular motion into reversible chiroptical function. *Nat Commun* 7:10591. doi: [10.1038/ncomms10591](https://doi.org/10.1038/ncomms10591)
16. Linko V, Eerikäinen M, Kostianen MA (2015) A modular DNA origami-based enzyme cascade nanoreactor. *Chem Commun* 51:5351–5354. doi: [10.1039/C4CC08472A](https://doi.org/10.1039/C4CC08472A)
17. Liu M, Fu J, Hejesen C, et al (2013) A DNA tweezer-actuated enzyme nanoreactor. *Nat Commun*. doi: [10.1038/ncomms3127](https://doi.org/10.1038/ncomms3127)
18. Williams S, Lund K, Lin C, et al (2009) Tiamat: A three-dimensional editing tool for complex DNA structures. In: Goel A, Simmel FC, Sos'vik P (eds) *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*. Springer Berlin Heidelberg, pp 90–101

19. Douglas SM, Marblestone AH, Teerapittayanon S, et al (2009) Rapid prototyping of 3D DNA-origami shapes with caDNAno. *Nucleic Acids Res* 37:5001–5006. doi: 10.1093/nar/gkp436
20. Bai X -c., Martin TG, Scheres SHW, Dietz H (2012) Cryo-EM structure of a 3D DNA-origami object. *Proc Natl Acad Sci* 109:20012–20017. doi: 10.1073/pnas.1215713109
21. Kim DN, Kilchherr F, Dietz H, Bathe M (2012) Quantitative prediction of 3D solution shape and flexibility of nucleic acid nanostructures. *Nucleic Acids Res* 40:2862–2868. doi: 10.1093/nar/gkr1173
22. Pan K, Kim D-N, Zhang F, et al (2014) Lattice-free prediction of three-dimensional structure of programmed DNA assemblies. *Nat Commun* 5:5578. doi: 10.1038/ncomms6578
23. Sedeh RS, Pan K, Adendorff MR, et al (2016) Computing Nonequilibrium Conformational Dynamics of Structured Nucleic Acid Assemblies. *J Chem Theory Comput* 12:261–273. doi: 10.1021/acs.jctc.5b00965
24. Doye JPK, Ouldridge TE, Louis AA, et al (2013) Coarse-graining DNA for simulations of DNA nanotechnology. *Phys Chem Chem Phys* 15:20395. doi: 10.1039/c3cp53545b
25. Snodin BEK, Romano F, Rovigatti L, et al (2016) Direct Simulation of the Self-Assembly of a Small DNA Origami. *ACS Nano* 10:1724–1737. doi: 10.1021/acs.nano.5b05865
26. Yoo J, Aksimentiev A (2013) In situ structure and dynamics of DNA origami determined through molecular dynamics simulations. *Proc Natl Acad Sci* 110:20099–20104. doi: 10.1073/pnas.1316521110
27. Li C-Y, Hemmig EA, Kong J, et al (2015) Ionic Conductivity, Structural Deformation, and Programmable Anisotropy of DNA Origami in Electric Field. *ACS Nano* 9:1420–1433. doi: 10.1021/nn505825z
28. Slone SM, Li C-Y, Yoo J, Aksimentiev A (2016) Molecular mechanics of DNA bricks: in situ structure, mechanical properties and ionic conductivity. *New J Phys* 18:55012. doi: 10.1088/1367-2630/18/5/055012
29. Göpfrich K, Li C-Y, Mames I, et al (2016) Ion Channels Made from a Single Membrane-Spanning DNA Duplex. *Nano Lett* 16:4665–4669. doi: 10.1021/acs.nanolett.6b02039
30. Göpfrich K, Li CY, Ricci M, et al (2016) Large-Conductance Transmembrane Porin Made from DNA Origami. *ACS Nano* 10:8207–8214. doi: 10.1021/acs.nano.6b03759
31. Maffeo C, Yoo J, Aksimentiev A (2016) De novo reconstruction of DNA origami structures through atomistic molecular dynamics simulation. *Nucleic Acids Res* 44:3013–3019. doi: 10.1093/nar/gkw155
32. Phillips JC, Braun R, Wang W, et al (2005) Scalable molecular dynamics with NAMD. *J Comput Chem* 26:1781–1802. doi: 10.1002/jcc.20289

33. Humphrey W, Dalke A, Schulten K (1996) VMD: Visual molecular dynamics. *J Mol Graph* 14:33–38. doi: 10.1016/0263-7855(96)00018-5
34. VMD User Guide. <http://www.ks.uiuc.edu/Research/vmd/current/ug/ug.html>.
35. VMD Tutorial. <http://www.ks.uiuc.edu/Training/Tutorials/vmd/tutorialhtml/index.html>.
36. NAMD User Guide. <http://www.ks.uiuc.edu/Research/namd/current/ug/>.
37. NAMD Tutorial. <http://www.ks.uiuc.edu/Training/Tutorials/namd/namd-tutorial-unix-html/index.html>.
38. Yoo J, Aksimentiev A (2012) Improved parametrization of Li⁺, Na⁺, K⁺, and Mg²⁺ ions for all-atom molecular dynamics simulations of nucleic acid systems. *J Phys Chem Lett* 3:45–50. doi: 10.1021/jz201501a
39. Yoo J, Aksimentiev A (2012) Competitive Binding of Cations to Duplex DNA Revealed through Molecular Dynamics Simulations. *J Phys Chem B* 116:12946–12954. doi: 10.1021/jp306598y
40. Yoo J, Aksimentiev A (2016) The structure and intermolecular forces of DNA condensates. *Nucl Acids Res* 44:2036–2046. doi: 10.1093/nar/gkw081
41. Yoo J, Aksimentiev A (2015) Molecular Dynamics of Membrane-Spanning DNA Channels: Conductance Mechanism, Electro-Osmotic Transport, and Mechanical Gating. *J Phys Chem Lett* 6:4680–4687. doi: 10.1021/acs.jpcllett.5b01964