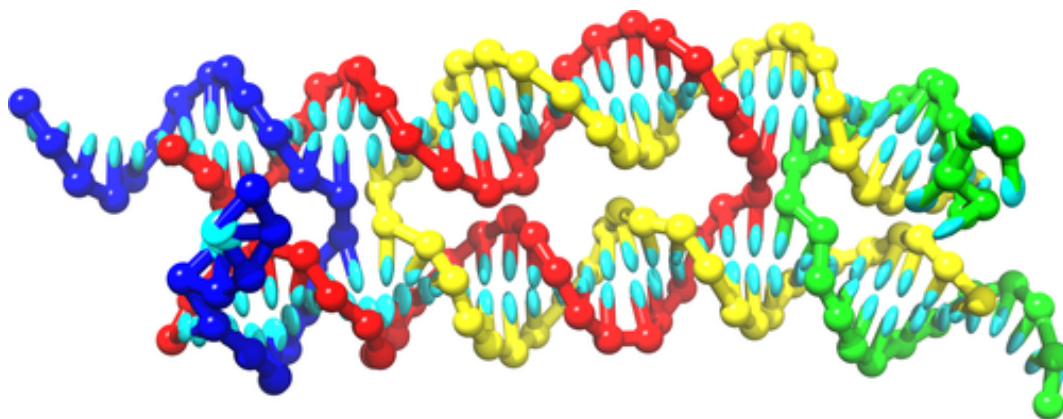


# Cadnano

## From Dna

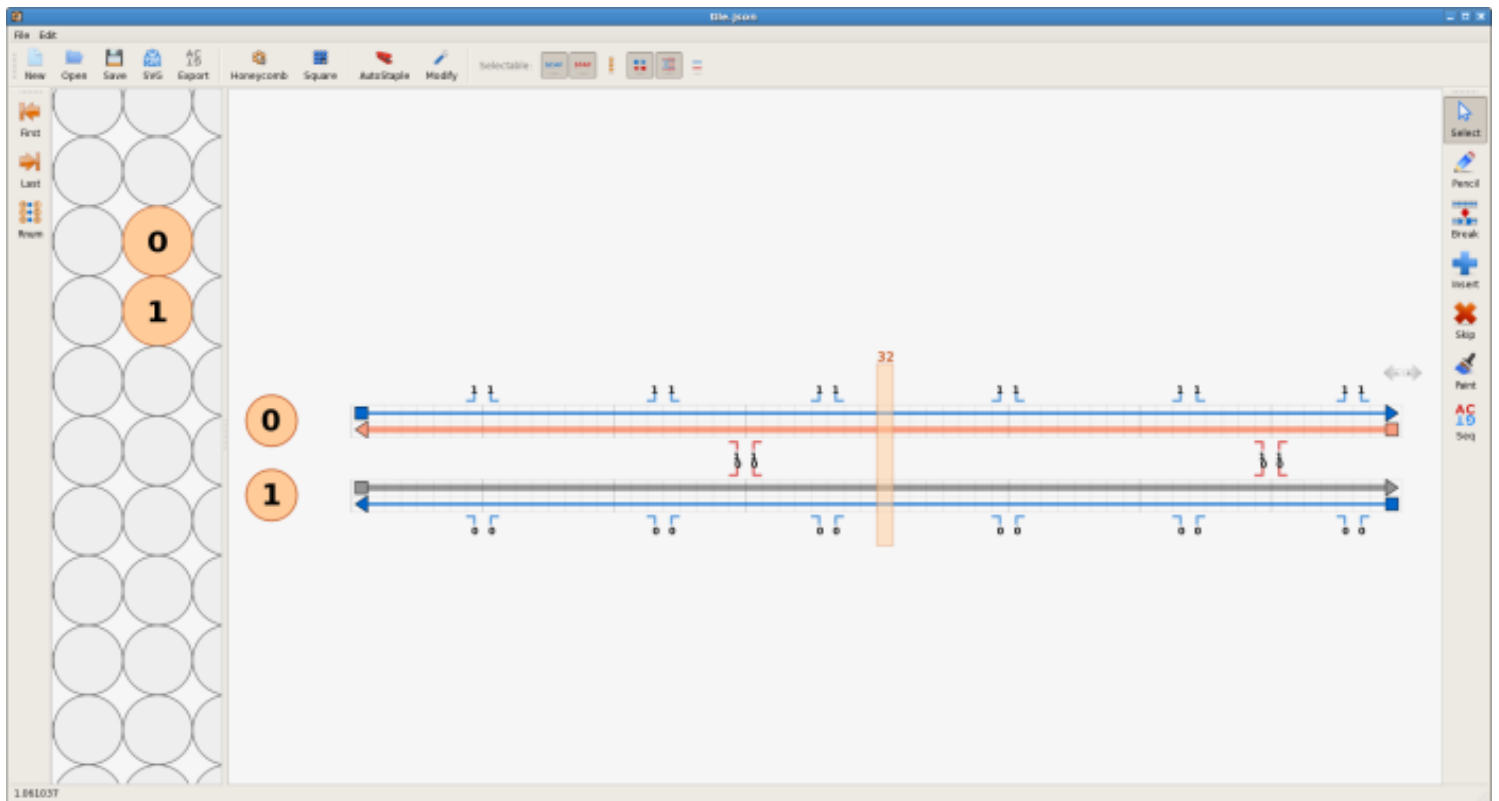
Cadnano (<http://cadnano.org>) is a tool for designing DNA origami structures. oxDNA includes an interface that allows origami designs generated in cadnano to be used as starting configurations for simulation. Cadnano can also be used in this way to make non-origami structures such as DNA tiles for use in the model.

## Example: DX tile

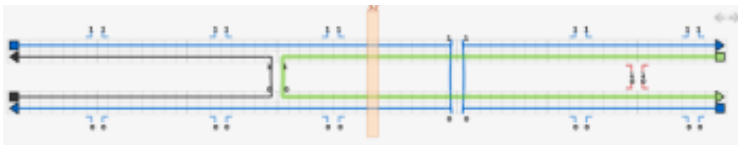


The use of cadnano for the purpose of creating starting configurations will be illustrated using a tile similar to the DX tile. In this case cadnano 2 will be used to design the tile, although the original cadnano operates in a similar way and is fully compatible with the interface to oxDNA. Note that, while using cadnano, any unwanted actions can be undone with ctrl+z (this feature is new in cadnano 2).

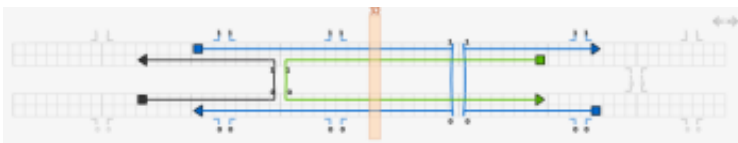
First click the blue "add new square lattice" button on the toolbar - a new square lattice will be created in the lattice view. The circles represent a cross-sectional view of potential DNA double helices on the square lattice. Create two empty virtual double helices in the path view by clicking on any two adjacent circles, one on top of the other. Next, using the pencil tool from the toolbar on the right, fill in all four rows in the path view by dragging from one end of each row to the other. Each double helix has two rows, and each row represents one of the strands of a double helix.



The next step is to add the crossovers between the double helices. Using origami terminology, the thin blue lines represent the scaffold strands, while the thicker lines of different colour represent the staple strands. Make a crossover between the staple strands near the middle of the design, by first clicking on one of the staple strands near its middle and then clicking one of the numbers near the middle next to that staple strand. Also click the number just to the left or right of the crossover you just made, to make a second crossover. Do the same for the scaffold strands - the crossovers can go either to the left or the right of the staple crossovers, not at the closest place to the staple crossovers, but the next closest place.



The tile is almost complete; the final step is to correct the lengths of the strands. Switch to the select tool and drag the ends of the strands to change their lengths. The two strands that are complementary to a double crossover should extend 8 bases past that crossover. The other two strands should extend 5 bases past those two strands. The yellow bar can help with tasks like this; drag it to any column to see the index of that column. If extra space is required, the virtual helices can be extended in either direction by clicking on the arrows at the top right of the topmost virtual helix in the path view.



The tile design is now finished and ready to be used as a starting configuration for simulation after some processing. The design can be found at [\\${oxDNA}/EXAMPLES/CADNANO\\_INTERFACE/TILE/tile.json](#).

## Using Cadnano Designs as Starting Configurations for oxDNA

The script `cadnano_interface.py` is used to generate an oxDNA configuration and topology file from a cadnano design. In addition it creates some files containing information about the origami that can be useful later for analysing its trajectory. Its usage is

```
python cadnano_interface.py <cadnano_file> <design_type> [box_size]
```

The design type must be either `sq` or `hc`, corresponding to either a square or honeycomb lattice - typically a 2D origami is on a square lattice, while a 3D origami is on a honeycomb lattice. The box size option allows the user to specify a simulation box size different to the default value of two times (in linear dimension) the largest dimension of the cadnano design.

When the configuration and topology files are first created, they cannot be used in an ordinary oxDNA simulation until they have been relaxed. This is achieved using an oxDNA MD simulation with a very low temperature and a very strongly coupled thermostat. An example input file for this simulation is found at `/${oxDNA}/EXAMPLES/CADNANO_INTERFACE/input_relax`. The process is very fast as the simulation need only run for around 100 steps.

For the tile discussed above, the configuration files before and after relaxation, as well as the topology file, can be found in `/${oxDNA}/EXAMPLES/CADNANO_INTERFACE/TILE/`.

## Limitations

The majority of structures without forced crossovers should work. However there are limitations:

For very large cadnano designs, or for designs of small origamis but with very long unused sections of virtual double helices, the script requires a very large amount of memory which may cause problems.

A large number of insertions or deletions on a small section of dna (where a section is defined as the bases between nodes, a node being a crossover or the end of a strand) may create a structure that cannot be relaxed.

Forced crossovers, that is crossovers made by the pencil tool instead of using the default cadnano default positions, will probably create a structure that cannot be relaxed.

Very large structures with very frequent crossovers may create a structure that cannot be relaxed.

Retrieved from "<http://dna.physics.ox.ac.uk/index.php/Cadnano>"

Category: Examples

- 
- This page was last modified 11:37, 16 May 2012.

## User's Guide

[Homepage](#)

[About This Resource](#)

[Examples & Applications](#)

[User's Guide](#)

[Community Portal](#)

[Release Notes](#)

[Feedback](#)

[Wiki](#)

For more information about this resource, please contact:  
[mark.bathe\\_at\\_MIT.EDU](mailto:mark.bathe_at_MIT.EDU)



Like

14

Send

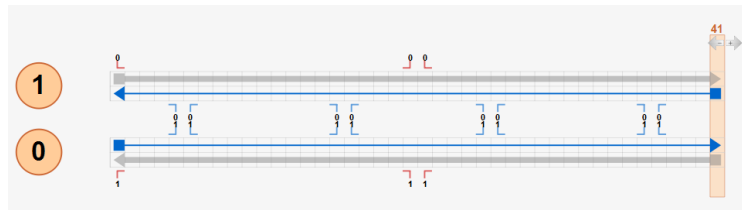
[File Formats](#)

[Tutorial](#)

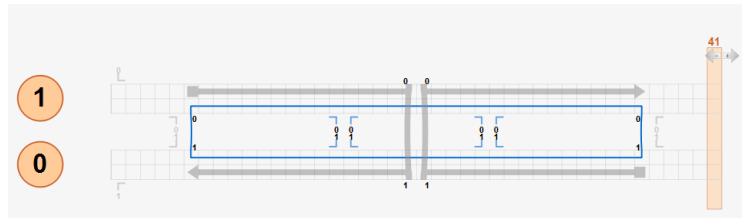
[Modeling Guide](#)

### 1. Crossovers between neighboring helices

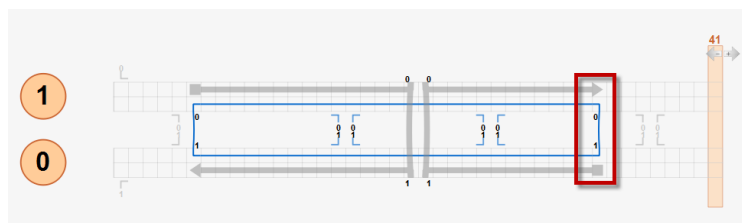
CanDo assumes that all crossovers between neighboring helices in the design are located at their natural positions defined in the caDNAno at which torsional mismatch between neighboring helices is minimal. These natural crossover positions can be seen in the caDNAno design panel when a strand is clicked as shown in the figure below.



Currently, users must follow this crossover rule to design DNA origami structures for CanDo analysis. Below is an example two-helix bundle design that follows the crossover rule whose 3D solution shape is expected to be straight.

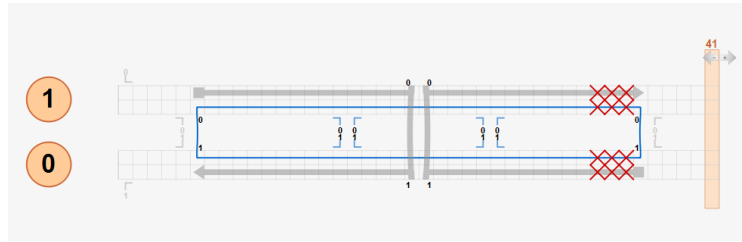


If a design does not follow the crossover rule, unexpected results may be obtained from the CanDo analysis. For example, in the modified two-helix design below, the right scaffold crossover is enforced at the position that is three basepairs left from its natural position as highlighted in the red box. Although there is torsional mismatch between basepairs connected by this crossover, expected to result in left-handed twist, CanDo predicts the straight solution shape because all crossovers are treated as natural ones.

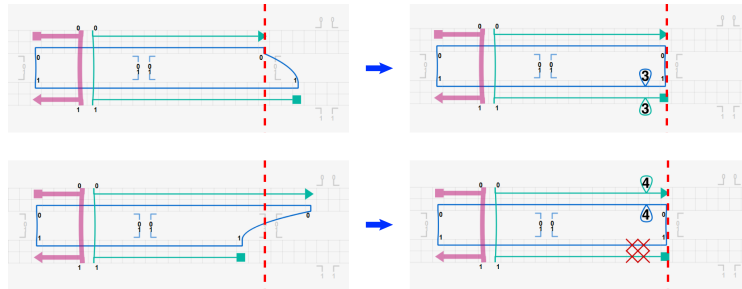


To simulate the effect of this mismatch properly, the design should be modified using insertions or deletions with natural crossovers as

below. Three deletions at each helix are added to the design while using natural crossovers only. CanDo predicts left-handed twist on the right half of the bundle correctly as expected.



Two more example design modifications are shown in the figure below where red dashed lines indicate natural crossover positions.



**An algorithm to account for the mismatch due to unnatural crossovers is currently under development.**

## 2. Crossovers between non-neighboring helices (Distant crossovers)

## User's Guide

**Homepage**

**About This Resource**

**Examples & Applications**

**User's Guide**

**Community Portal**

**Release Notes**

**Feedback**

**Wiki**

For more information about this resource, please contact:  
[mark.bathe\\_at\\_MIT.EDU](mailto:mark.bathe_at_MIT.EDU)



Like

14

Send

**File Formats**

**Tutorial**

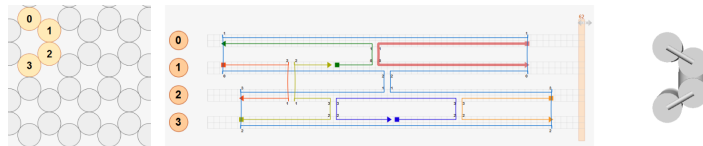
**Modeling Guide**

### 1. Crossovers between neighboring helices

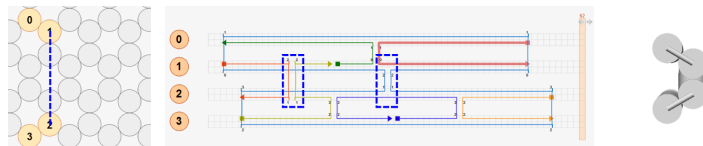
### 2. Crossovers between non-neighboring helices (Distant crossovers)

In wireframe structure design, crossovers between non-neighboring helices are used to connect substructures that are initially distant to each other but should be adjacent in the final relaxed configuration. CanDo defines these crossovers as distant crossovers that gradually shrink during the analysis until their length reduces to the distance between neighboring helices or the helix diameter. However, relative orientations of helices connected by distant crossovers remain unconstrained so that CanDo analysis is limited at present to on-lattice modeling of DNA origami structures. Thus, careful use of distance crossovers is advisable.

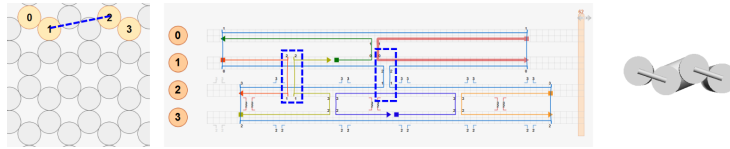
To illustrate this, let's consider three test designs of four-helix bundle that have the same connectivity map. The first design consists of four neighboring helices with crossovers at the natural positions. Because all basepairs are aligned at crossovers, the deformed shape (right) is the same as the initial layout (left).



In the second design, the entire structure is divided into two sub-bundles (helices 0-1 and helices 2-3) separated vertically in the initial layout. Two sub-bundles are connected by distant crossovers between helix 1 and helix 2 at the same axial positions used in the first design so that basepairs connected by these distant crossovers are already torsionally aligned (highlighted with blue dashed lines). Hence the same deformed shape is obtained as in the first design.



In the last design, helices 2-3 are placed in an arbitrary position. Basepairs connected by distant crossovers between helix 1 and helix 2 are not torsionally aligned. In addition, all crossovers between helix 2 and helix 3 violate the natural crossover rule. In this case, the deformed shape predicted by CanDo will be different from that of the first and second designs.



**We are currently developing a computational model to allow for more general crossovers, enabling lattice-free DNA origami design. Stay tuned!**