Automated, constraint-based analysis of tethered DNA nanostructures (SUPPORTING INFORMATION)

Matthew R. Lakin*[1,2,3] and Andrew Phillips[4]

[1]Department of Chemical & Biological Engineering, University of New Mexico, NM, USA
[2]Department of Computer Science, University of New Mexico, NM, USA
[3]Center for Biomedical Engineering, University of New Mexico, NM, USA
[4]Microsoft Research, 21 Station Rd, Cambridge CB1 2FB, UK

June 23, 2017

Table of Contents for Supporting Information

S1 Model details for first stator binding to second, in the straight-line system

In this section, we present model details for the example system derived from [1], where the first stator (which is opened, and with a fuel molecule bound) is attempting to bind to the *second* stator (which is still closed), in a system with three stators arranged in a straight line.

*Corresponding author (`mlakin@cs.unm.edu`)

Here, and in all examples, we used a floating-point representation for our constraint variables with an 8 bit exponent and an 8 bit significand, and the following values for other system parameters:

- $\mathsf{L}_{\mathsf{ds}} = 0.34\,\text{nm}$,
- $\mathsf{L}_{\mathsf{ss}} = 0.68\,\text{nm}$, and
- $\varepsilon = 10^{-5}$.

For the purposes of displaying the examples in this document, we used:

- inter-stator distance = 1 nm,
- length of the `lsp` domain = 5 nucleotides,
- length of the `tsp` domain = 5 nucleotides,
- length of the toehold domains = 5 nucleotides, and
- length of the recognition domains = 30 nucleotides.

Note that, in this example, we added extra constraints that all *y*-coordinates equal zero as this decreases solving time.

S1.1 Input representation of the localized process

The domains used in this, and all subsequent examples, are as follows:

- `a0`, a toehold of length 5 nucleotides
- `x`, a toehold of length 5 nucleotides
- `y`, a toehold of length 5 nucleotides
- `s`, a recognition domain of length 30 nucleotides
- `lsp`, a recognition domain of length 5 nucleotides
- `tsp`, a recognition domain of length 5 nucleotides

The input code for this system is as follows:

```
( a0^!i1 s!i2
| s!i4 y^!i3 loopspacer s*!i2 a0^*!i1 tetherspacer tether(0,0)
| y^*!i3 s*!i4 loopspacer x^!i5 s!i6
| s y^ loopspacer s*!i6 x^*!i5 tetherspacer tether(1,0)
| s!i7 y^ loopspacer s*!i7 x^* tetherspacer tether(2,0) )
```

S1.2 Fully explicit representation of the localized process

Here, and elsewhere in this document, we use the `%` symbol to represent the position variables introduced automatically by the prototype compiler where they were not supplied in the input code. Furthermore, we use the `@` symbol to represent the *length* of a domain, so `s@30` is a domain `s` with length 30 (this enables the results of the condensing process to be confirmed, see below).

```
( <(x%1, y%1, z%1) a0^!i1@5 (x%2, y%2, z%2) s!i2@30 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s!i4@30 (x%5, y%5, z%5) y^!i3@5 (x%6, y%6, z%6) loopspacer@5
   (x%7, y%7, z%7) s*!i2@30 (x%8, y%8, z%8) a0^*!i1@5 (x%9, y%9, z%9)
   tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) y^*!i3@5 (x%11, y%11, z%11) s*!i4@30 (x%12, y%12, z%12) loopspacer@5
   (x%13, y%13, z%13) x^!i5@5 (x%14, y%14, z%14) s!i6@30 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s@30 (x%17, y%17, z%17) y^@5 (x%18, y%18, z%18) loopspacer@5
   (x%19, y%19, z%19) s*!i6@30 (x%20, y%20, z%20) x^*!i5@5 (x%21, y%21, z%21)
   tetherspacer@5 tether(1.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y^@5 (x%24, y%24, z%24) loopspacer@5
   (x%25, y%25, z%25) s*!i7@30 (x%26, y%26, z%26) x^*@5 (x%27, y%27, z%27)
   tetherspacer@5 tether(2.0, 0.0, 0.0)> )
```

S1.3 Fully condensed localized process

Here, and elsewhere in this document, we use the hyphen symbol to connect individual domains to represent the combined domain produced by the condensing procedure. So, for example, the condensed combination of domains `s` and `y` is written as `s-y` and its complement as `s-y*`.

```
( <(x%1, y%1, z%1) a0-s!i1@35 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s-y!i4@35 (x%6, y%6, z%6) loopspacer@5 (x%7, y%7, z%7) a0-s*!i1@35
   (x%9, y%9, z%9) tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) s-y*!i4@35 (x%12, y%12, z%12) loopspacer@5 (x%13, y%13, z%13)
   x-s!i5@35 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s-y-loopspacer@40 (x%19, y%19, z%19) x-s*!i5@35 (x%21, y%21, z%21)
   tetherspacer@5 tether(1.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y-loopspacer@10 (x%25, y%25, z%25)
   s*!i7@30 (x%26, y%26, z%26) x-tetherspacer@10 tether(2.0, 0.0, 0.0)> )
```

S1.4 Derived set of constraints

The total number of constraints in this example is: 107

```
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 >= 1.1015625*(2**7)
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 <= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 >= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 <= 1.1015625*(2**7)
x%7 - x%6 * x%7 - x%6 + y%7 - y%6 * y%7 - y%6 + z%7 - z%6 * z%7 - z%6 <= 1.4375*(2**3)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 >= 1.1015625*(2**7)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 <= 1.1015625*(2**7)
+0.0 - x%9 * +0.0 - x%9 + +0.0 - y%9 * +0.0 - y%9 + +0.0 - z%9 * +0.0 - z%9 <= 1.4375*(2**3)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 >= 1.1015625*(2**7)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 <= 1.1015625*(2**7)
x%13 - x%12 * x%13 - x%12 + y%13 - y%12 * y%13 - y%12 + z%13 - z%12 * z%13 - z%12 <= 1.4375*(2**3)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 >= 1.1015625*(2**7)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 <= 1.1015625*(2**7)
x%19 - x%16 * x%19 - x%16 + y%19 - y%16 * y%19 - y%16 + z%19 - z%16 * z%19 - z%16 <= 1.4375*(2**9)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 >= 1.1015625*(2**7)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 <= 1.1015625*(2**7)
1 - x%21 * 1 - x%21 + +0.0 - y%21 * +0.0 - y%21 + +0.0 - z%21 * +0.0 - z%21 <= 1.4375*(2**3)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 >= 1.625*(2**6)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 <= 1.625*(2**6)
x%25 - x%23 * x%25 - x%23 + y%25 - y%23 * y%25 - y%23 + z%25 - z%23 * z%25 - z%23 <= 1.4375*(2**5)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 >= 1.625*(2**6)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 <= 1.625*(2**6)
1*(2**1) - x%26 * 1*(2**1) - x%26 + +0.0 - y%26 * +0.0 - y%26 + +0.0 - z%26 * +0.0 - z%26 <= 1.4375*(2**5)
x%1 == x%9
y%1 == y%9
z%1 == z%9
x%7 == x%3
y%7 == y%3
z%7 == z%3
x%13 == x%21
y%13 == y%21
z%13 == z%21
x%19 == x%15
y%19 == y%15
z%19 == z%15
x%4 == x%12
```

```
y%4 == y%12
z%4 == z%12
x%10 == x%6
y%10 == y%6
z%10 == z%6
x%22 == x%26
y%22 == y%26
z%22 == z%26
x%25 == x%23
y%25 == y%23
z%25 == z%23
z%1 >= -1.3046875*(2**-17)
z%3 >= -1.3046875*(2**-17)
z%4 >= -1.3046875*(2**-17)
z%6 >= -1.3046875*(2**-17)
z%7 >= -1.3046875*(2**-17)
z%9 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%10 >= -1.3046875*(2**-17)
z%12 >= -1.3046875*(2**-17)
z%13 >= -1.3046875*(2**-17)
z%15 >= -1.3046875*(2**-17)
z%16 >= -1.3046875*(2**-17)
z%19 >= -1.3046875*(2**-17)
z%21 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%22 >= -1.3046875*(2**-17)
z%23 >= -1.3046875*(2**-17)
z%25 >= -1.3046875*(2**-17)
z%26 >= -1.3046875*(2**-17)
+0.0 == +0.0
y%1 >= -1.3046875*(2**-17)
y%1 <= 1.3046875*(2**-17)
y%3 >= -1.3046875*(2**-17)
y%3 <= 1.3046875*(2**-17)
y%4 >= -1.3046875*(2**-17)
y%4 <= 1.3046875*(2**-17)
y%6 >= -1.3046875*(2**-17)
y%6 <= 1.3046875*(2**-17)
y%7 >= -1.3046875*(2**-17)
y%7 <= 1.3046875*(2**-17)
y%9 >= -1.3046875*(2**-17)
y%9 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
y%10 >= -1.3046875*(2**-17)
y%10 <= 1.3046875*(2**-17)
y%12 >= -1.3046875*(2**-17)
y%12 <= 1.3046875*(2**-17)
y%13 >= -1.3046875*(2**-17)
y%13 <= 1.3046875*(2**-17)
y%15 >= -1.3046875*(2**-17)
y%15 <= 1.3046875*(2**-17)
y%16 >= -1.3046875*(2**-17)
```

```
y%16 <= 1.3046875*(2**-17)
y%19 >= -1.3046875*(2**-17)
y%19 <= 1.3046875*(2**-17)
y%21 >= -1.3046875*(2**-17)
y%21 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
y%22 >= -1.3046875*(2**-17)
y%22 <= 1.3046875*(2**-17)
y%23 >= -1.3046875*(2**-17)
y%23 <= 1.3046875*(2**-17)
y%25 >= -1.3046875*(2**-17)
y%25 <= 1.3046875*(2**-17)
y%26 >= -1.3046875*(2**-17)
y%26 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
```

S2 Model details for first stator binding to third, in the straight-line system

In this section, we present model details for the example system derived from [1], where the first stator (which is opened, and with a fuel molecule bound) is attempting to bind to the *third* stator (which is still closed), in a system with three stators arranged in a straight line. Note that, in this example, we added extra constraints that all *y*-coordinates equal zero as this decreases solving time.

S2.1 Input representation of the localized process

```
( a0^!i1 s!i2
| s!i4 y^!i3 loopspacer s*!i2 a0^*!i1 tetherspacer tether(0,0)
| y^*!i3 s*!i4 loopspacer x^!i5 s!i6
| s y^ loopspacer s*!i6 x^*!i5 tetherspacer tether(2,0)
| s!i7 y^ loopspacer s*!i7 x^* tetherspacer tether(1,0) )
```

S2.2 Fully explicit representation of the localized process

```
( <(x%1, y%1, z%1) a0^!i1@5 (x%2, y%2, z%2) s!i2@30 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s!i4@30 (x%5, y%5, z%5) y^!i3@5 (x%6, y%6, z%6) loopspacer@5
   (x%7, y%7, z%7) s*!i2@30 (x%8, y%8, z%8) a0^*!i1@5 (x%9, y%9, z%9)
   tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) y^*!i3@5 (x%11, y%11, z%11) s*!i4@30 (x%12, y%12, z%12)
   loopspacer@5 (x%13, y%13, z%13) x^!i5@5 (x%14, y%14, z%14) s!i6@30 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s@30 (x%17, y%17, z%17) y^@5 (x%18, y%18, z%18) loopspacer@5
   (x%19, y%19, z%19) s*!i6@30 (x%20, y%20, z%20) x^*!i5@5 (x%21, y%21, z%21)
   tetherspacer@5 tether(2.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y^@5 (x%24, y%24, z%24) loopspacer@5
   (x%25, y%25, z%25) s*!i7@30 (x%26, y%26, z%26) x^*@5 (x%27, y%27, z%27)
   tetherspacer@5 tether(1.0, 0.0, 0.0)> )
```

S2.3 Fully condensed localized process

```
( <(x%1, y%1, z%1) a0-s!i1@35 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s-y!i4@35 (x%6, y%6, z%6) loopspacer@5 (x%7, y%7, z%7) a0-s*!i1@35
   (x%9, y%9, z%9) tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) s-y*!i4@35 (x%12, y%12, z%12) loopspacer@5 (x%13, y%13, z%13)
   x-s!i5@35 (x%15, y%15, z%15)>
```

```
| <(x%16, y%16, z%16) s-y-loopspacer@40 (x%19, y%19, z%19) x-s*!i5@35 (x%21, y%21, z%21)
   tetherspacer@5 tether(2.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y-loopspacer@10 (x%25, y%25, z%25)
   s*!i7@30 (x%26, y%26, z%26) x-tetherspacer@10 tether(1.0, 0.0, 0.0)> )
```

S2.4 Derived set of constraints

The total number of constraints in this example is: 107

```
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 >= 1.1015625*(2**7)
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 <= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 >= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 <= 1.1015625*(2**7)
x%7 - x%6 * x%7 - x%6 + y%7 - y%6 * y%7 - y%6 + z%7 - z%6 * z%7 - z%6 <= 1.4375*(2**3)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 >= 1.1015625*(2**7)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 <= 1.1015625*(2**7)
+0.0 - x%9 * +0.0 - x%9 + +0.0 - y%9 * +0.0 - y%9 + +0.0 - z%9 * +0.0 - z%9 <= 1.4375*(2**3)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 >= 1.1015625*(2**7)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 <= 1.1015625*(2**7)
x%13 - x%12 * x%13 - x%12 + y%13 - y%12 * y%13 - y%12 + z%13 - z%12 * z%13 - z%12 <= 1.4375*(2**3)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 >= 1.1015625*(2**7)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 <= 1.1015625*(2**7)
x%19 - x%16 * x%19 - x%16 + y%19 - y%16 * y%19 - y%16 + z%19 - z%16 * z%19 - z%16 <= 1.4375*(2**9)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 >= 1.1015625*(2**7)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 <= 1.1015625*(2**7)
1*(2**1) - x%21 * 1*(2**1) - x%21 + +0.0 - y%21 * +0.0 - y%21 + +0.0 - z%21 * +0.0 - z%21 <= 1.4375*(2**3)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 >= 1.625*(2**6)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 <= 1.625*(2**6)
x%25 - x%23 * x%25 - x%23 + y%25 - y%23 * y%25 - y%23 + z%25 - z%23 * z%25 - z%23 <= 1.4375*(2**5)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 >= 1.625*(2**6)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 <= 1.625*(2**6)
1 - x%26 * 1 - x%26 + +0.0 - y%26 * +0.0 - y%26 + +0.0 - z%26 * +0.0 - z%26 <= 1.4375*(2**5)
x%1 == x%9
y%1 == y%9
z%1 == z%9
x%7 == x%3
y%7 == y%3
z%7 == z%3
x%13 == x%21
y%13 == y%21
z%13 == z%21
x%19 == x%15
y%19 == y%15
z%19 == z%15
x%4 == x%12
y%4 == y%12
z%4 == z%12
x%10 == x%6
y%10 == y%6
z%10 == z%6
x%22 == x%26
y%22 == y%26
z%22 == z%26
x%25 == x%23
y%25 == y%23
```

```
z%25 == z%23
z%1 >= -1.3046875*(2**-17)
z%3 >= -1.3046875*(2**-17)
z%4 >= -1.3046875*(2**-17)
z%6 >= -1.3046875*(2**-17)
z%7 >= -1.3046875*(2**-17)
z%9 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%10 >= -1.3046875*(2**-17)
z%12 >= -1.3046875*(2**-17)
z%13 >= -1.3046875*(2**-17)
z%15 >= -1.3046875*(2**-17)
z%16 >= -1.3046875*(2**-17)
z%19 >= -1.3046875*(2**-17)
z%21 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%22 >= -1.3046875*(2**-17)
z%23 >= -1.3046875*(2**-17)
z%25 >= -1.3046875*(2**-17)
z%26 >= -1.3046875*(2**-17)
+0.0 == +0.0
y%1 >= -1.3046875*(2**-17)
y%1 <= 1.3046875*(2**-17)
y%3 >= -1.3046875*(2**-17)
y%3 <= 1.3046875*(2**-17)
y%4 >= -1.3046875*(2**-17)
y%4 <= 1.3046875*(2**-17)
y%6 >= -1.3046875*(2**-17)
y%6 <= 1.3046875*(2**-17)
y%7 >= -1.3046875*(2**-17)
y%7 <= 1.3046875*(2**-17)
y%9 >= -1.3046875*(2**-17)
y%9 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
y%10 >= -1.3046875*(2**-17)
y%10 <= 1.3046875*(2**-17)
y%12 >= -1.3046875*(2**-17)
y%12 <= 1.3046875*(2**-17)
y%13 >= -1.3046875*(2**-17)
y%13 <= 1.3046875*(2**-17)
y%15 >= -1.3046875*(2**-17)
y%15 <= 1.3046875*(2**-17)
y%16 >= -1.3046875*(2**-17)
y%16 <= 1.3046875*(2**-17)
y%19 >= -1.3046875*(2**-17)
y%19 <= 1.3046875*(2**-17)
y%21 >= -1.3046875*(2**-17)
y%21 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
y%22 >= -1.3046875*(2**-17)
y%22 <= 1.3046875*(2**-17)
y%23 >= -1.3046875*(2**-17)
```

```
y%23 <= 1.3046875*(2**-17)
y%25 >= -1.3046875*(2**-17)
y%25 <= 1.3046875*(2**-17)
y%26 >= -1.3046875*(2**-17)
y%26 <= 1.3046875*(2**-17)
+0.0 >= -1.3046875*(2**-17)
+0.0 <= 1.3046875*(2**-17)
```

S3 Model details for first stator binding to second, in the 90° angle system

In this section, we present model details for the example system derived from [1], where the first stator (which is opened, and with a fuel molecule bound) is attempting to bind to the *second* stator (which is still closed), in a system with three stators arranged to form a 90° angle. Note that, in this example, we *did not* add extra constraints that all *y*-coordinates equal zero, as our stators are not located in a straight line.

S3.1 Input representation of the localized process

```
( a0^!i1 s!i2
| s!i4 y^!i3 loopspacer s*!i2 a0^*!i1 tetherspacer tether(0,0)
| y^*!i3 s*!i4 loopspacer x^!i5 s!i6
| s y^ loopspacer s*!i6 x^*!i5 tetherspacer tether(1,0)
| s!i7 y^ loopspacer s*!i7 x^* tetherspacer tether(1,1) )
```

S3.2 Fully explicit representation of the localized process

```
( <(x%1, y%1, z%1) a0^!i1@5 (x%2, y%2, z%2) s!i2@30 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s!i4@30 (x%5, y%5, z%5) y^!i3@5 (x%6, y%6, z%6) loopspacer@5
   (x%7, y%7, z%7) s*!i2@30 (x%8, y%8, z%8) a0^*!i1@5 (x%9, y%9, z%9)
   tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) y^*!i3@5 (x%11, y%11, z%11) s*!i4@30 (x%12, y%12, z%12)
   loopspacer@5 (x%13, y%13, z%13) x^!i5@5 (x%14, y%14, z%14) s!i6@30 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s@30 (x%17, y%17, z%17) y^@5 (x%18, y%18, z%18) loopspacer@5
   (x%19, y%19, z%19) s*!i6@30 (x%20, y%20, z%20) x^*!i5@5 (x%21, y%21, z%21)
   tetherspacer@5 tether(1.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y^@5 (x%24, y%24, z%24) loopspacer@5
   (x%25, y%25, z%25) s*!i7@30 (x%26, y%26, z%26) x^*@5 (x%27, y%27, z%27)
   tetherspacer@5 tether(1.0, 1.0, 0.0)> )
```

S3.3 Fully condensed localized process

```
( <(x%1, y%1, z%1) a0-s!i1@35 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s-y!i4@35 (x%6, y%6, z%6) loopspacer@5 (x%7, y%7, z%7) a0-s*!i1@35
   (x%9, y%9, z%9) tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) s-y*!i4@35 (x%12, y%12, z%12) loopspacer@5 (x%13, y%13, z%13)
   x-s!i5@35 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s-y-loopspacer@40 (x%19, y%19, z%19) x-s*!i5@35 (x%21, y%21, z%21)
   tetherspacer@5 tether(1.0, 0.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y-loopspacer@10 (x%25, y%25, z%25)
   s*!i7@30 (x%26, y%26, z%26) x-tetherspacer@10 tether(1.0, 1.0, 0.0)> )
```

S3.4 Derived set of constraints

The total number of constraints in this example is: 67

```
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 >= 1.1015625*(2**7)
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 <= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 >= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 <= 1.1015625*(2**7)
x%7 - x%6 * x%7 - x%6 + y%7 - y%6 * y%7 - y%6 + z%7 - z%6 * z%7 - z%6 <= 1.4375*(2**3)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 >= 1.1015625*(2**7)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 <= 1.1015625*(2**7)
+0.0 - x%9 * +0.0 - x%9 + +0.0 - y%9 * +0.0 - y%9 + +0.0 - z%9 * +0.0 - z%9 <= 1.4375*(2**3)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 >= 1.1015625*(2**7)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 <= 1.1015625*(2**7)
x%13 - x%12 * x%13 - x%12 + y%13 - y%12 * y%13 - y%12 + z%13 - z%12 * z%13 - z%12 <= 1.4375*(2**3)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 >= 1.1015625*(2**7)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 <= 1.1015625*(2**7)
x%19 - x%16 * x%19 - x%16 + y%19 - y%16 * y%19 - y%16 + z%19 - z%16 * z%19 - z%16 <= 1.4375*(2**9)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 >= 1.1015625*(2**7)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 <= 1.1015625*(2**7)
1 - x%21 * 1 - x%21 + +0.0 - y%21 * +0.0 - y%21 + +0.0 - z%21 * +0.0 - z%21 <= 1.4375*(2**3)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 >= 1.625*(2**6)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 <= 1.625*(2**6)
x%25 - x%23 * x%25 - x%23 + y%25 - y%23 * y%25 - y%23 + z%25 - z%23 * z%25 - z%23 <= 1.4375*(2**5)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 >= 1.625*(2**6)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 <= 1.625*(2**6)
1 - x%26 * 1 - x%26 + 1 - y%26 * 1 - y%26 + +0.0 - z%26 * +0.0 - z%26 <= 1.4375*(2**5)
x%1 == x%9
y%1 == y%9
z%1 == z%9
x%7 == x%3
y%7 == y%3
z%7 == z%3
x%13 == x%21
y%13 == y%21
z%13 == z%21
x%19 == x%15
y%19 == y%15
z%19 == z%15
x%4 == x%12
y%4 == y%12
z%4 == z%12
x%10 == x%6
y%10 == y%6
z%10 == z%6
x%22 == x%26
y%22 == y%26
z%22 == z%26
x%25 == x%23
y%25 == y%23
z%25 == z%23
z%1 >= -1.3046875*(2**-17)
z%3 >= -1.3046875*(2**-17)
z%4 >= -1.3046875*(2**-17)
z%6 >= -1.3046875*(2**-17)
z%7 >= -1.3046875*(2**-17)
z%9 >= -1.3046875*(2**-17)
+0.0 == +0.0
```

```
z%10 >= -1.3046875*(2**-17)
z%12 >= -1.3046875*(2**-17)
z%13 >= -1.3046875*(2**-17)
z%15 >= -1.3046875*(2**-17)
z%16 >= -1.3046875*(2**-17)
z%19 >= -1.3046875*(2**-17)
z%21 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%22 >= -1.3046875*(2**-17)
z%23 >= -1.3046875*(2**-17)
z%25 >= -1.3046875*(2**-17)
z%26 >= -1.3046875*(2**-17)
+0.0 == +0.0
```

S4 Model details for first stator binding to third, in the 90° angle system

In this section, we present model details for the example system derived from [1], where the first stator (which is opened, and with a fuel molecule bound) is attempting to bind to the *third* stator (which is still closed), in a system with three stators arranged to form a 90° angle. Note that, in this example, we *did not* add extra constraints that all *y*-coordinates equal zero, as our stators are not located in a straight line.

S4.1 Input representation of the localized process

```
( a0^!i1 s!i2
| s!i4 y^!i3 loopspacer s*!i2 a0^*!i1 tetherspacer tether(0,0)
| y^*!i3 s*!i4 loopspacer x^!i5 s!i6
| s y^ loopspacer s*!i6 x^*!i5 tetherspacer tether(1,1)
| s!i7 y^ loopspacer s*!i7 x^* tetherspacer tether(1,0) )
```

S4.2 Fully explicit representation of the localized process

```
( <(x%1, y%1, z%1) a0^!i1@5 (x%2, y%2, z%2) s!i2@30 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s!i4@30 (x%5, y%5, z%5) y^!i3@5 (x%6, y%6, z%6) loopspacer@5
   (x%7, y%7, z%7) s*!i2@30 (x%8, y%8, z%8) a0^*!i1@5 (x%9, y%9, z%9)
   tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) y^*!i3@5 (x%11, y%11, z%11) s*!i4@30 (x%12, y%12, z%12)
   loopspacer@5 (x%13, y%13, z%13) x^!i5@5 (x%14, y%14, z%14) s!i6@30 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s@30 (x%17, y%17, z%17) y^@5 (x%18, y%18, z%18) loopspacer@5
   (x%19, y%19, z%19) s*!i6@30 (x%20, y%20, z%20) x^*!i5@5 (x%21, y%21, z%21)
   tetherspacer@5 tether(1.0, 1.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y^@5 (x%24, y%24, z%24) loopspacer@5
   (x%25, y%25, z%25) s*!i7@30 (x%26, y%26, z%26) x^*@5 (x%27, y%27, z%27)
   tetherspacer@5 tether(1.0, 0.0, 0.0)> )
```

S4.3 Fully condensed localized process

```
( <(x%1, y%1, z%1) a0-s!i1@35 (x%3, y%3, z%3)>
| <(x%4, y%4, z%4) s-y!i4@35 (x%6, y%6, z%6) loopspacer@5 (x%7, y%7, z%7)
   a0-s*!i1@35 (x%9, y%9, z%9) tetherspacer@5 tether(0.0, 0.0, 0.0)>
| <(x%10, y%10, z%10) s-y*!i4@35 (x%12, y%12, z%12) loopspacer@5
   (x%13, y%13, z%13) x-s!i5@35 (x%15, y%15, z%15)>
| <(x%16, y%16, z%16) s-y-loopspacer@40 (x%19, y%19, z%19) x-s*!i5@35
   (x%21, y%21, z%21) tetherspacer@5 tether(1.0, 1.0, 0.0)>
| <(x%22, y%22, z%22) s!i7@30 (x%23, y%23, z%23) y-loopspacer@10
   (x%25, y%25, z%25) s*!i7@30 (x%26, y%26, z%26) x-tetherspacer@10
```

```
tether(1.0, 0.0, 0.0)> )
```

S4.4 Derived set of constraints

The total number of constraints in this example is: 67

```
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 >= 1.1015625*(2**7)
x%3 - x%1 * x%3 - x%1 + y%3 - y%1 * y%3 - y%1 + z%3 - z%1 * z%3 - z%1 <= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 >= 1.1015625*(2**7)
x%6 - x%4 * x%6 - x%4 + y%6 - y%4 * y%6 - y%4 + z%6 - z%4 * z%6 - z%4 <= 1.1015625*(2**7)
x%7 - x%6 * x%7 - x%6 + y%7 - y%6 * y%7 - y%6 + z%7 - z%6 * z%7 - z%6 <= 1.4375*(2**3)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 >= 1.1015625*(2**7)
x%9 - x%7 * x%9 - x%7 + y%9 - y%7 * y%9 - y%7 + z%9 - z%7 * z%9 - z%7 <= 1.1015625*(2**7)
+0.0 - x%9 * +0.0 - x%9 + +0.0 - y%9 * +0.0 - y%9 + +0.0 - z%9 * +0.0 - z%9 <= 1.4375*(2**3)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 >= 1.1015625*(2**7)
x%12 - x%10 * x%12 - x%10 + y%12 - y%10 * y%12 - y%10 + z%12 - z%10 * z%12 - z%10 <= 1.1015625*(2**7)
x%13 - x%12 * x%13 - x%12 + y%13 - y%12 * y%13 - y%12 + z%13 - z%12 * z%13 - z%12 <= 1.4375*(2**3)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 >= 1.1015625*(2**7)
x%15 - x%13 * x%15 - x%13 + y%15 - y%13 * y%15 - y%13 + z%15 - z%13 * z%15 - z%13 <= 1.1015625*(2**7)
x%19 - x%16 * x%19 - x%16 + y%19 - y%16 * y%19 - y%16 + z%19 - z%16 * z%19 - z%16 <= 1.4375*(2**9)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 >= 1.1015625*(2**7)
x%21 - x%19 * x%21 - x%19 + y%21 - y%19 * y%21 - y%19 + z%21 - z%19 * z%21 - z%19 <= 1.1015625*(2**7)
1 - x%21 * 1 - x%21 + 1 - y%21 * 1 - y%21 + +0.0 - z%21 * +0.0 - z%21 <= 1.4375*(2**3)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 >= 1.625*(2**6)
x%23 - x%22 * x%23 - x%22 + y%23 - y%22 * y%23 - y%22 + z%23 - z%22 * z%23 - z%22 <= 1.625*(2**6)
x%25 - x%23 * x%25 - x%23 + y%25 - y%23 * y%25 - y%23 + z%25 - z%23 * z%25 - z%23 <= 1.4375*(2**5)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 >= 1.625*(2**6)
x%26 - x%25 * x%26 - x%25 + y%26 - y%25 * y%26 - y%25 + z%26 - z%25 * z%26 - z%25 <= 1.625*(2**6)
1 - x%26 * 1 - x%26 + +0.0 - y%26 * +0.0 - y%26 + +0.0 - z%26 * +0.0 - z%26 <= 1.4375*(2**5)
x%1 == x%9
y%1 == y%9
z%1 == z%9
x%7 == x%3
y%7 == y%3
z%7 == z%3
x%13 == x%21
y%13 == y%21
z%13 == z%21
x%19 == x%15
y%19 == y%15
z%19 == z%15
x%4 == x%12
y%4 == y%12
z%4 == z%12
x%10 == x%6
y%10 == y%6
z%10 == z%6
x%22 == x%26
y%22 == y%26
z%22 == z%26
x%25 == x%23
y%25 == y%23
z%25 == z%23
z%1 >= -1.3046875*(2**-17)
z%3 >= -1.3046875*(2**-17)
```

```
z%4 >= -1.3046875*(2**-17)
z%6 >= -1.3046875*(2**-17)
z%7 >= -1.3046875*(2**-17)
z%9 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%10 >= -1.3046875*(2**-17)
z%12 >= -1.3046875*(2**-17)
z%13 >= -1.3046875*(2**-17)
z%15 >= -1.3046875*(2**-17)
z%16 >= -1.3046875*(2**-17)
z%19 >= -1.3046875*(2**-17)
z%21 >= -1.3046875*(2**-17)
+0.0 == +0.0
z%22 >= -1.3046875*(2**-17)
z%23 >= -1.3046875*(2**-17)
z%25 >= -1.3046875*(2**-17)
z%26 >= -1.3046875*(2**-17)
+0.0 == +0.0
```

References

[1] Muscat RA, Strauss K, Ceze L, Seelig G. DNA-based Molecular Architecture with Spatially Localized Components. In: Proceedings of ISCA 13; 2013. .