Hybrid Systems Analysis of *DNA Supercoiling Regulation*

Raaghav Thirumaligai
Department of Mechanical Engineering, UCSB rthirumaligai@ucsb.edu

***Abstract*—In this report, we analyze a Hybrid Automaton description for the Supercoiling of DNA. DNA has continuous behavior coming from the mechanical stresses and discrete jumps induced by enzyme binding events. We derive dwell-time conditions on enzyme reset times that guarantee weak forward invariance and global attractivity of a target supercoiling set, ensuring DNA stability near a homeostatic set point. Numerical simulations validate the theoretical results and highlight the essential role of Topoisomerase in maintaining supercoiling levels. This work provides a mathematical framework connecting biological mechanisms with hybrid control theory, offering insights into DNA regulatory dynamics based on ideas from [1].**

I. Introduction

When DNA exists in a cell, it's state of supercoiling (σ) is carefully regulated by enzymes to maintain an ideal amount of gene expression. We will consider the ideal supercoiling state of the DNA to be inherently unstable since it may want to uncoil towards a state of neutral coiling ($\sigma = 0$). With the feedback control of two enzymes, the supercoiling state is regulated to the homeostatic set point σ^*. Based on the work done in [2] we suggest a model for DNA regulation where a Gyrase enzyme removes some positive coiling δ_g (twists the DNA in a negative direction), and a Topoisomerase enzyme adds some positive coiling δ_t (twists the DNA in a positive direction). We will look at the supercoiling coordinate, x, shifted so that 0 is the ideal state of coiling,

$$x := \sigma - \sigma^*,$$

and Fig. 1 gives a sketch of the system.

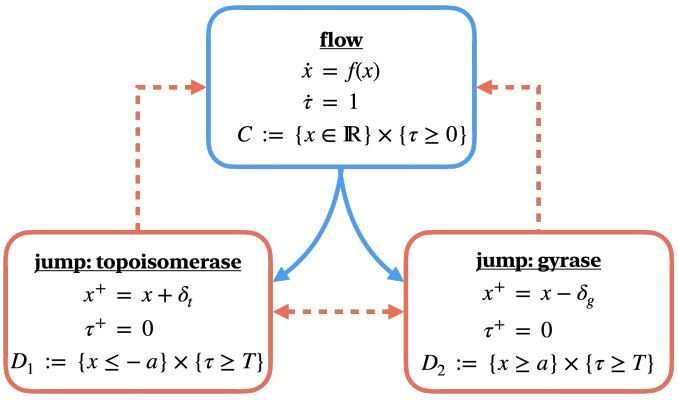

Fig. 1. Hybrid Automaton Diagram illustrating the components and interactions within the hybrid system model.

II. System Description

A. Continuous Dynamics

The DNA will try to uncoil towards the neutral state at some rate K.

$$\dot{x} = -K(x + \sigma^*); \quad \forall x \in C$$

Our flow-set is just any value $x \in \mathbb{R}$ since the dynamics are always valid. We need to add a dwell time constraint representing some physicality where the DNA is responding to a change or the enzyme is resetting.

$$\dot{\tau} = 1; \quad \forall \tau \in C$$

Time is always increasing and we can flow for any time as well so the flow set has $t \in \mathbb{R}$

B. Discrete Dynamics

There are two sets of discrete behavior and assuming the coiling on the segment we observe only has space for one enzyme at a time, we have a shared timer.

$$\tau^+ = 0; \quad \forall \tau \in D$$

The jump set for the timer includes all time after T representing some relaxation time for the enzyme/DNA. For the coiling, the first jump set will trigger a Topoisomerase while the second is for the Gyrase. The idea is that when coiling deviates beyond some threshold, the respective enzyme binding affinity allows it to bind and alter the coiling with a discrete change.

$$x^+ = x + \delta_t; \quad \forall x \in D_1$$

$$x^+ = x - \delta_g; \quad \forall x \in D_2$$

C. Hybrid Model

$$\mathcal{H}: \begin{cases} \dot{x} = -K(x + \sigma^*) & x \in \mathbb{R} \\ \dot{\tau} = 1 & \tau \in [0, \infty) \\ x^+ = \begin{cases} x + \delta_t & x \in (-\infty, -a] \\ x - \delta_g & x \in [a, \infty) \end{cases} \\ \tau^+ = 0 & \tau \in [T, \infty) \end{cases} \tag{1}$$

D. Hybrid Basic Conditions On The Data

(A1) C and D are closed subsets of $\mathbb{R}^n$.
- $\rightarrow$ $C = \mathbb{R} \times \mathbb{R}$ which is closed in $\mathbb{R}^n$
- $\rightarrow$ $D = (\{x \leq -a\} \cup \{x \geq a\}) \times (\tau \in [T, \infty))$ also is closed in $\mathbb{R}^n$

(A2) $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is outer semicontinuous and locally bounded relative to C, $C \subseteq \operatorname{dom} F$, and $F(x)$ is convex for every $x \in C$.
- $\rightarrow$ $F(x, \tau) = \begin{bmatrix} -K(x + \sigma^*) \\ 1 \end{bmatrix}$ which is continuous in C and locally bounded. This is singleton so convex $\forall$ x in C.

(A3) $G : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is outer semicontinuous and locally bounded relative to D, and $D \subseteq \operatorname{dom} G$.
- $\rightarrow$ $G(x, \tau)$ is single-valued and continuous on D, so its graph is closed and outer-semicontinuous.

III. Analysis

A. Equilibria / Desired Set

a) Equilibrium of the flow map.: When the state evolves inside the flow set C (without any enzymes binding) the dynamics reduce to

$$\dot{x} = -K(x + \sigma^*), \qquad \dot{\tau} = 1.$$

So we have an equilibrium of:

$$(x_e, \tau) \;=\; (-\sigma^*,\, \tau \to \infty).$$

Because we only allow flows while $|x| < a$ (or just after a jump), and the biologically relevant parameter range satisfies $\sigma^* > a$, this equilibrium lies **outside** the admissible flow region. Admittedly we might jump to this equilibrium point but, it is not stable since after a short time T, another enzyme will bind, and we will jump. There are conditions where trajectories converge arbitrarily close to the equilibrium for large enzyme reset times, but the state will always jump away for finite T. We will find requirements on T to ensure every viable trajectory therefore reaches the jump set D before it deviates too far and converges to (x_e, τ); making flow equilibrium is unattainable, which is desirable biologically (DNA never drifts to the mechanically neutral state and stays within a of the homeostatic set point σ^*).

b) Desired hybrid set: We define the ideal set

$$\mathcal{A} := \{(x, \tau) \;:\; |x| \leq a,\; \tau \geq 0\},$$

noting that a should be greater than $\frac{\max\{\delta_t, \delta_g\}}{2}$ since we cannot design a forward invariant set if we can jump out of it.

B. Dwell–time condition on T

The timer length $T > 0$ must be small enough s.t. $F(x, \tau)$ cannot carry the state out of the desired range $|x| \leq a$ before a jump is enabled. With $\dot{x} = -K(x + \sigma^*)$ the flow solution is

$$x(t) \;=\; (x_0 + \sigma^*)\, e^{-Kt} - \sigma^*, \qquad 0 \leq t \leq T,$$

where $x_0 \in (-a, a)$ is the post–jump value. Since $x(t)$ is always decreasing within $\mathcal{A}$, we need to ensure that the jump in x due to the enzyme is greater than the amount that x will flow from the lower edge of our set; ensuring our enzyme makes progress to restore the state back to the set. Really, this is not enough since we want the state to move within the set and also stay within the set. This requirement means we should impose an even stricter bound: the enzyme must reset faster than the time it takes the state to drift from $(-a + \delta_t)$ back to the threshold $-a$.

Solving $(x(T) = -a;\, x_0 = -a + \delta_t)$ for $T_{\sup}$ gives

$$-a \;=\; (-a + \delta_t + \sigma^*)\, e^{-KT} \;-\; \sigma^*$$

$$\rightarrow \quad e^{-KT} = \frac{\sigma^* - a}{\sigma^* - a + \delta_t},$$

and hence

$$T_{\sup} := \frac{-1}{K} \ln\!\left(\frac{\sigma^* - a}{\sigma^* - a + \delta_t}\right).$$

Because the drift is monotone, any $T \in [0, T_{\sup})$ guarantees the state cannot exit $\mathcal{A}$ before the next jump.

a) Design guideline.: Choose the enzymatic reset time T to satisfy

$$0 \leq T < T_{\sup}.$$

When this bound holds, every flow segment stays inside C and each complete solution experiences infinitely many jumps, so the Lyapunov analysis in Sec. III guarantees **global asymptotic stability** of the target set $\mathcal{A}$. Biologically, $T_{\sup}$ corresponds to some minimum Topoisomerase concentration and $(T = 0)$ is just an abundance, meaning if the enzymes are available, they can bind. An interesting thing to note in our analysis is that $T_{\sup}$ is independent of the Gyrase. We are motivated to see if global asymptotic stability is achievable in the system and if we can achieve GASness without Gyrase.

If $T > T_{\sup}$ the DNA can drift past $-a$ while $\tau < T$, thereby breaking positive invariance of the target set $\mathcal{A}$ and preventing stability of this set.

C. Lyapunov Candidate

Trying the classic energy Lyapunov function:

$$V(x) \;:=\; \frac{1}{2}\, x^2, \qquad (x, \tau) \in C \cup D.$$

Decrease along flows: For $|x| < a$ we have

$$\dot{V} = \frac{1}{2}(2x)(\dot{x}) \tag{2}$$

$$\dot{V} = x(-K(x + \sigma^*)) \tag{3}$$

$$\text{check} \rightarrow 0 > \dot{V} = -K(x^2) - K\sigma^* x \tag{4}$$

$$0 > -K(x)(x - \sigma^*) \tag{5}$$

$$0 < (x)(x - \sigma^*) \tag{6}$$

we can see that there is a region $x \in (0, \sigma^*)$ where the inequality does not hold, so our Lyapunov function is increasing. This is a region we have ensured will converge to the set $\mathcal{A}$ by requiring an appropriate value for T.

D. Forward Invariance of $\mathcal{A}$

To have forward invariance of $\mathcal{A}$ we need solutions that start in $\mathcal{A}$ to stay within $\mathcal{A}$ forever. This is not true, and we can easily illustrate with an example: Take the initial condition on the boundary of the set with $\tau_0 = 0$. Clearly only flow can occur, and we will immediately leave the set. We can look for weak forward invariance, and it is freely given if we consider the initial condition modification to $\tau_0 > T$. This allows us to immediately jump further back into the set and flow until we reach the boundary again.

TABLE I
MODEL PARAMETERS

Symbol	Value	Description
k	3	Mechanical Restoration
a	0.05	Ideal set radius (enzymes don't bind)
δ_t	0.095	Jump: add positive coils
δ_g	-0.09	Jump: remove positive coils
σ^*	0.095	Homeostatic set-point (Clark *et al.*)
T	$T_{\text{sup}} - 0.01 = 0.3562$	Time delay for enzyme reset

IV. Simulation Results

Using the Matlab Hybrid Systems Toolbox, we can simulate our automaton. In table I the system values are given, either to model a biological constraint or convergence constraint. Simulating with $\tau_0 = 0$.

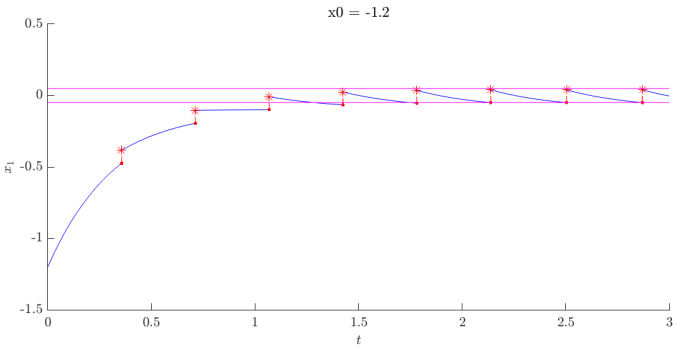

Fig. 2. Coiling trajectory from negatively coiled initial condition. The pink lines show the ideal set $\mathcal{A}$

V. Discussion

The role of Topoisomerase in DNA regulation is clearly necessary for this model. As shown in Fig. 2, the hybrid system rapidly drives a negatively coiled initial state into the target set $\mathcal{A}$ using a combination of discrete jumps (red markers) and continuous flows (blue arcs). Local trajectories in Fig. 3 illustrate how errors originating from a bundle of initial states collapse into $\mathcal{A}$ within a few cycles. Removing Gyrase function (Fig. 4) still yields robust convergence, indicating that Topoisomerase alone suffices to maintain set invariance. However, defunctionalizing Topoisomerase (Fig. 5) abolishes convergence and causes trajectories to converge toward the neutral state $x = -\sigma^*$, confirming Topoisomerase's essential role.

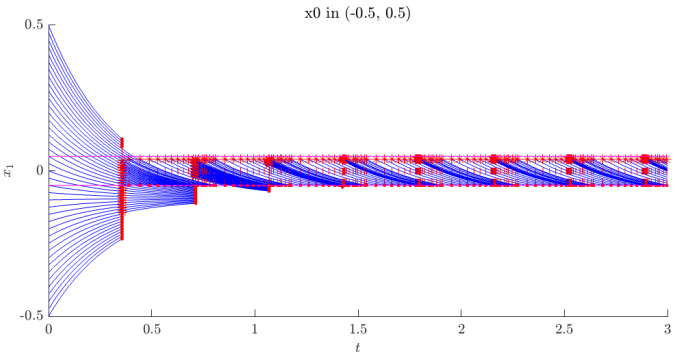

Fig. 3. 40 Coiling trajectories from local initial conditions. Gyrase is only used on a few of these trajectories, and we would still arrive to $\mathcal{A}$ in finite time without this due to the exponentially decaying dynamics.

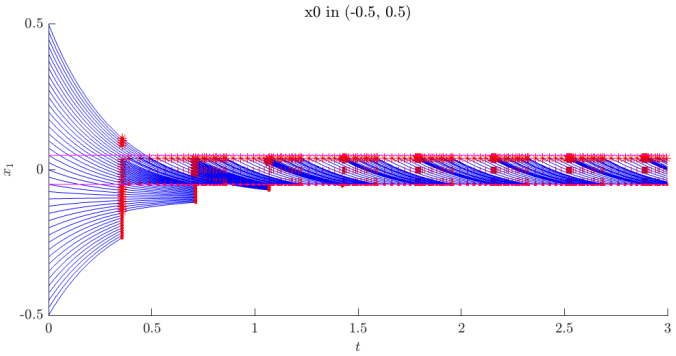

Fig. 4. Defunctionalizing Gyrase (allowing it to jump but not remove any coils), we see that there is still convergence to the set.

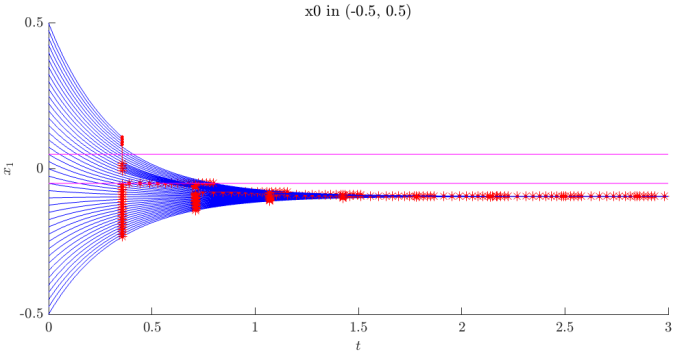

Fig. 5. Defunctionalizing Topoisomerase (still using δ_t for T_{sup}), we lose set convergence. The trajectories converge asymptotically to the state of neutral coiling at $x = -\sigma^* = -0.095$

VI. Conclusion

- We have proved forward invariance and demonstrated global attractivity of the hybrid automaton's target set $\mathcal{A}$ through both analytical dwell-time bounds and the structure of jumps and flows which are validated in numerical simulations.
- Figures 2 and 3 highlight rapid convergence from both extreme and local initial conditions.
- While Gyrase removal (Fig. 4) does not prevent convergence, Topoisomerase removal (Fig. 5) breaks set

invariance, underscoring the necessity of this enzyme to maintain a more positively coiled state.

- Future work can explore parameter robustness, experimental estimation of δ_t, δ_g, T, and extensions to stochastic enzyme kinetics and multi-segment DNA models.

References

[1] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*, 1st ed. Princeton, NJ, USA: Princeton University Press, 2012.

[2] H. Clark, E. Yeung, R. Thirumaligai, and C. Giusto, "Topoisomerase-based control of cellular transcription and growth," *bioRxiv*, 2025. [Online]. Available: https://doi.org/10.1101/2025.05.26.655193

Appendix

```
classdef supercoil_switch < HybridSystem
    % Hybrid model for DNA supercoiling
        ↪ regulation via discrete
    % gyrase / topoisomerase unwindings /
        ↪ windings.


    %% tunable parameters
    properties
        k   = 3;            % open-loop drift
            ↪ rate  (dot{x} = k*x)
        a   = 0.05;         % guard radius
            ↪ acceptable state of coiling /
            ↪ enzymes cant bind
        top = 0;%.09;         % add pos coils
            ↪ (jump step)
        gyr = -.09;        % remove pos coils
            ↪ (jump step)
        xSP = 0.095;        % homeostatic
            ↪ setpoint from Clark et al.
             % time delay -> enzyme reset in
                ↪ some sense
    end

    %% fixed indices (immutable)
    properties ( SetAccess = immutable )
        x_index = 1;       % supercoil state
        t_index = 2;       % enzyme timer
        q_index = 3;       % logic flag
    end

    %% constructor
    methods
        function this = supercoil_switch()
            % State dimension = 2 (x and q)
            this@HybridSystem(3);
        end
    end

    %% Hybrid System Set and Map
        ↪ implementations
    methods
        function xdot = flowMap(this, x, ~, ~)
            ↪       % f(x)
            xval = x( this.x_index );
            xdot = [ - this.k * (xval + this.
                ↪ xSP); % unstable drift
                     1  ;
                        ↪
                        ↪ % the arrow of time
                        ↪  !!
                    0 ];
                        ↪
                        ↪ % dq/dt = 0 during
                        ↪ flow
        end

        function xplus = jumpMap(this, x)
            ↪           % g(x)
            xval = x( this.x_index );

            if  xval >=  this.a
                ↪                       %
                ↪ positively coiled above
                ↪ ideal set
```

```
                xval = xval + this.gyr;
            elseif xval <= -this.a
                ↪                   % negatively
                ↪  coiled below ideal set
                xval = xval + this.top;
            end
            qplus = 1 - x( this.q_index );
            tplus = 0;
            xplus = [ xval ; tplus; qplus ];
        end

        function inC = flowSetIndicator(this,
            ↪ x)  % C
            xval = abs( x( this.x_index ) );
            tval = x( this.t_index);

            inC = (xval >= 0) & (tval >= 0);
        end

        function inD = jumpSetIndicator(this,
            ↪ x)  % D
            xval = abs( x( this.x_index ) );
            tval = x( this.t_index ) ;
            Tsup = -1/this.k * log((this.xSP -
                ↪  this.a) / (this.xSP-this.a
                ↪ +0.09)) - 0.01;
            inD = ( xval >= this.a ) & (tval
                ↪ >=   Tsup );
        end
    end
end
```

Listing 1. Wrapped code example