ATP synthase: some surprising results from a rigid-body analysis

Fynn Tseng[1]
[1]Department of Bioengineering, University of Washington

1 Abstract

The energy of glucose's oxidation (in an aerobe) is stored within a proton gradient. It is thought that this gradient is dissipated across a large, amphipathic protein known as ATP synthase. For the most part, ATP synthase is a black box. Most attempts at modeling it are based on the compartments, or states, postulated in Boyer's binding-change mechanism. Recently, Y. Nartsissov and E. Mashkovtesa wrote a paper in which they used Newtonian mechanics to arrive at a physical model. In this effort, we shall recapitulate the authors' work, and consider the system's robustness. We show that by varying the initial conditions, quasi-limit cycles are obtained at several loci in the phase plane. These cycles are sensitive to the initial conditions (i.e. they are not true limit cycles) and to a major parameter in the model. These loci can be connected in one trace if special initial positions are chosen.

2 Introduction

The structure of ATP synthase was resolved over one decade ago (Stock et. al, 1999). The protein was determined to be composed of two major domains. The F_1 domain is in the cytosol, and it catalyzes ATP synthesis. In the F_0 domain, c_0 monomers are thought to form a transmembrane ring. Rotation of this ring is induced by a coulombic force between to Asp61 residue of c_0 and the Arg210 residue of a (Groth and Walker, 1997). The former accepts a proton and translocates it across the membrane. In current understanding, this transfer creates a conformation change in the F0 rotor, enabling it to phosphorylate ADP, bound in the active site (Rastogi and Girvin, 2004).

There are two prominent models of the dynamics of ATP synthase. The first (Schulten et al, 2004) is a stochastic model which describes the motion of the α-helices of each c_0 monomer, as well as the rotation of the entire ring. The flow of protons through F_0 is said to merely bias the rotation of the c_0 ring. A Langevin equation was used to model all random (Brownian) motion of each of the helices. There was no deterministic force acting upon each monomer, but rather, a potential contour into which the angles could fall in order to minimize non-binding strain. The authors found that the ring cannot rotate without the a- c_0 dipole. This was defended by steered molecular dynamics simulations.

In the second model (Gao et. al, 2003), the authors propose a six-state model of ATP synthesis. The enzyme kinetics (rate of synthesis as a function of substrate) are elucidated for two different isoforms of ATP synthase. Some parameters are taken from other experimental studies, and others are inferred (from hyperbolic kinetics). In their model, they show that there is an progressive loss of free energy as the F_1 domain rotates.

In the binding-change mechanism (Boyer et. al, 1997), each catalytic subunit of ATP synthase occupies one of three states. There is paucity of modeling that has gone into explaining the transitions between these states, or how the proton-motive force is transduced into cyclic motion, and how that motion acts to synthesize ATP. We know that the c_0 subunit undergoes a significant conformational shift when it engages in acid-base reactions (Boyer, 1999; Rastogi and Girvin, 2004),

but this structural data is not sufficient to explain, in a quanitative manner, the chemo-osmotic hypothesis.

In the present model, the enzyme is viewed as a fully deterministic system. There are no random forces acting upon it. The authors claim that this is an appropriate modeling choice since the enzyme is symmetric (and is thereby resistant to small and random collisions), and the observation that the catalytic function seems to depend, in large part, on some well-characterized, which arise for the structure. In particular, For each c_0 subunit, the electrostatic force (between the charged residues) supplies a time-variant torque. This causes rotation of the ring, and is opposed by two properties: the bilayer viscosity and the elasticity of the γ-stalk of F_1 (Nartsissov and Mashkovtesa, 2006).

3 The Model

The net torque of the enzyme is given by the sum of the viscous, elastic, and electrostatic moments. A positive convention is defined from the F_0 to F_1 (out of the matrix). Thus, we obtain a single non-linear ODE in polar coordinates. Θ is the angular position of one c0 peptide.

$$k_1 \ddot{\Theta} + k_2 \dot{\Theta} + k_3 \mathrm{E} \cdot \left\{ \Theta + k_4 \Theta_r \prod_{i=1}^{3} \left(1 - \Delta\left(k, i - \frac{n}{3} \right) \right) \right\} - k_5 \sin\{\alpha\Theta - \Phi\} = 0 \qquad \textbf{(1)}$$

Here, k_1 is the moment of inertia, k_2 is the damping coefficient, k_3 is the rotational analogue of the spring constant, and k_5 is the electrostatic attraction between the two charged residues discussed above. These parameters are lumped, but the physical constants are decomposed in TABLE 1. By assuming that the enzyme behaves as a rigid body, we take certain liberties.

Note that **(1)** is valid only for the period of one c_0 subunit. As the F0 ring rotates, the intial conditions must be reset, since a newly ionized $c_0$interacts with the a domain. Thus, only one subunit at a time is subject to the same torques that are stipulated above. We may rewrite the equation to reflect this fact:

$$k_1 \ddot{\Theta} = \sum_{i=0}^{n} \left\{ -k_2 \dot{\Theta} - k_3 \left[\Theta + (i \bmod 4)\Theta_r \right] + k_5 \sin(\alpha\Theta - \Phi) \right\} \left[u(t - (i-1)\tau) - u(t - i\tau) \right] \qquad \textbf{(2)}$$

where $\bar{u}(t) = u(t - (i-1)\tau) - u(t - i\tau)$ is the step function and τ is the period (the time required to reach Θ_r). We implemented this later form in an ODE solver by using the state-space conversion

$$\begin{pmatrix} \dot{z}_2 \\ \dot{z}_1 \end{pmatrix} = \sum_{i=1}^{n/3} \begin{pmatrix} -k_3 \left[z_1 + (i \bmod 4)\Theta_r \right] + k_5 \sin(\alpha\Theta - \Phi) & -k_2 z_1 \bar{u}(i,t) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \qquad \textbf{(3)}$$

The solvation is carried out in a piecewise manner.

Parameter	Description	Value
$I = m/R^2$	*Moment of inertia of the F_0 domain*	$m = 1.48 \cdot 10^{-22}$ kg, $R = 3.0$ nm
Θ, Θ_r	*Angular position of the c-domain charged residue, angular separation of each domain*	$\Theta_r = 2\pi/n$
α	*Normalized angular position*	$\pi - \frac{\Phi}{\Theta_r}$
Φ	*Initial position of c-domain charge*	-1.603
Δ	*Dummy parameter (reiteration of model for each ATP)*	Cryptic
q_1, q_2	*a-domain Arg-210 charge, c-domain Asp-61 charge*	$+1.6 \cdot 10^{-19}$, $-1.6 \cdot 10^{-19}$ C
$\varepsilon, \varepsilon_0$	*Dielectric constant of membrane, permittivity of free-space*	$2.0 \cdot 10^{-12}$ F/m, $8.9 \cdot 10^{-12}$ F/m
h	*Length of c-domain*	4.0 nm
η	*Viscosity of membrane*	10^{-3} Pa × s
R_γ, L_γ	*Radius and Length of γ domain*	0.8 nm, 6 nm
E_γ	*Young's module of γ domain*	Varied
σ	*Poisson ration of γ domain*	0.4
Initial Condition		**Value**
$\Theta(0)$	Initial position	Varied
$\dot{\Theta}(0)$	The final angular velocity of the previous solution (recurrent)	ω_k

4 Results

We were able to reproduce the reported time-domain responses of the angular position (Nartissov and Mashkovtesa, 2006). The simulation shown in FIGURE 1 (a-c) ends after the synthesis of one ATP. As expected, the initial c_0 subunit has traveled 120°. The behavior is continuous (since the boundary conditions are equated at every iteration), but non-differentiable at $t = i \cdot \tau$. We have implicitly modeled the proton transfer by stopping the solution at the end of one period. We obtained isomorphic results for the simulation of the torque due to viscosity, although there is a discrepancy of 10^{-7}. This could be explained by using physical constants that were too small.

To study the stability of the system, it was necessary to run the simulation for longer times. To this effect, we assumed that the elastic torque kept increasing. Since the electrostatic force is bounded and periodic, this would have no effect on the behavior. By simulating four entire cycles for various initial positions spanning $[-2\pi, 2\pi]$ in intervals of $2\Theta_r$, we found that the response is rather complex, as seen in FIGURE 2 (a-c). Indeed, each different initial value renders a unique phase portrait. Further, for $\Theta_r > 2\pi$, the phase space is not modular, as we would expect. When we used larger initial values of Θ_r, we saw new cycles appearing at more distant locations. Although the distribution of these equilibria seem symmetric about $\Theta_r = 0$, the cycles are not symmetric between each other. We suppose that the non-modularity and asymmetry is caused by the effect of elastic torque, a linear function of Θ. Since we were remissive about removing this torque after four cycles, the period of the solution is naturally perturbed.

The full solutions to **(3)**, under various initial conditions, are varied, as in FIGURE 2 (a-d). When the initial angles begin far from zero, the behavior is clearly non-sinusoidal, in contrast to

FIGURE 1 (d). We obtained several different classes of quasi-limit cycles, which seem to occur at fixed points. Some of these cycles are rather ordinary (nearly sinusoidal), whereas others are fairly exotic. As the initial position becomes large (and consequently, the initial torque of the stalk), we obtain cycles that look like those seen with the canonical Van der Pol oscillator, as in FIGURE 2 (b). Contrary to being the convergent solution to a finite range of initial conditions (with fixed parameters), these cycles are dependent on the initial conditions, and seem to be unique to each one.

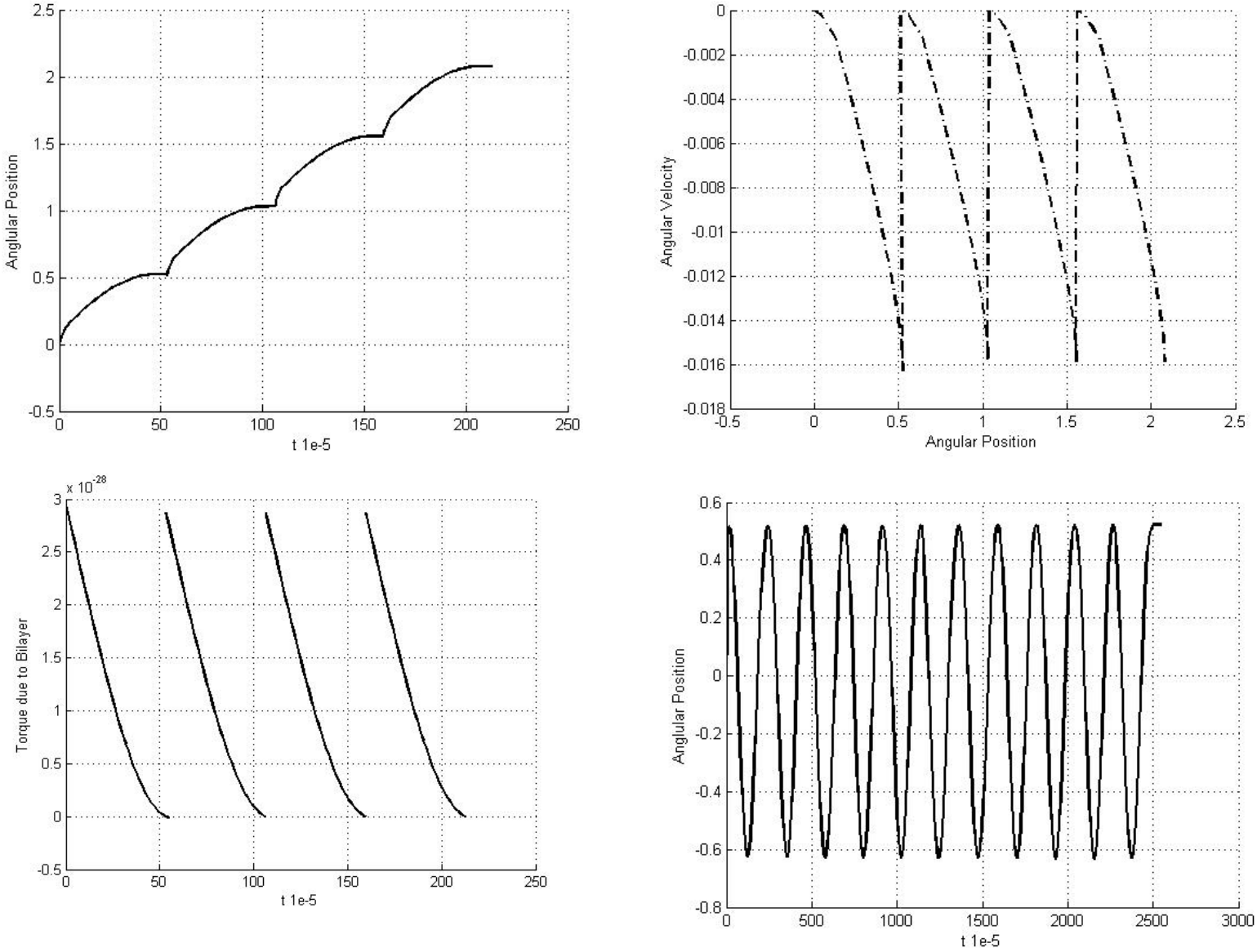
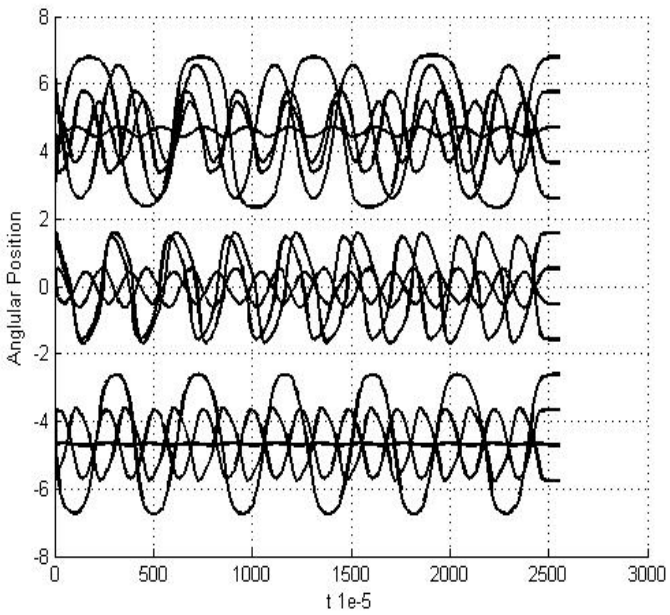


FIGURE 1 (a) Time response of the angular position for the initial conditions $\Theta(0)=0$; $\dot{\Theta}(0)=\omega_k$ only a partial cycle is shown. (b) The phase response of the solution in (a). (c) The toque produced by the membrane; discontinuities occur when the velocity. (d) Full time response for the initial condition vector listed in (a). The solution looks sinusoidal, but this is due to the initial conditions.



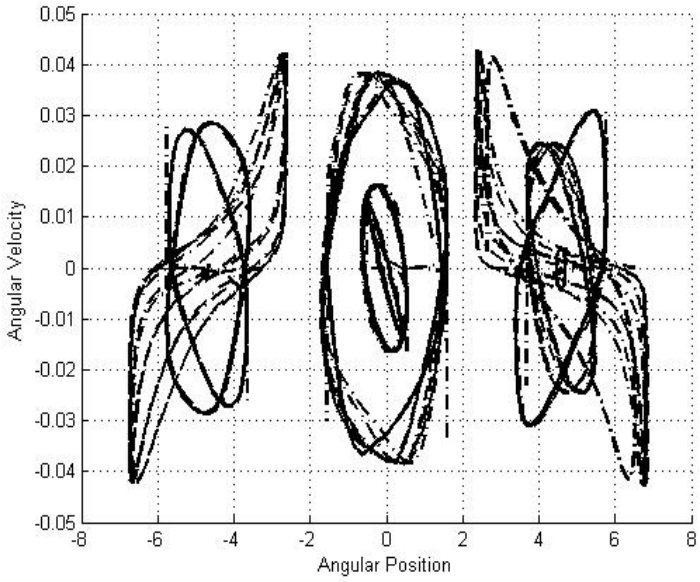
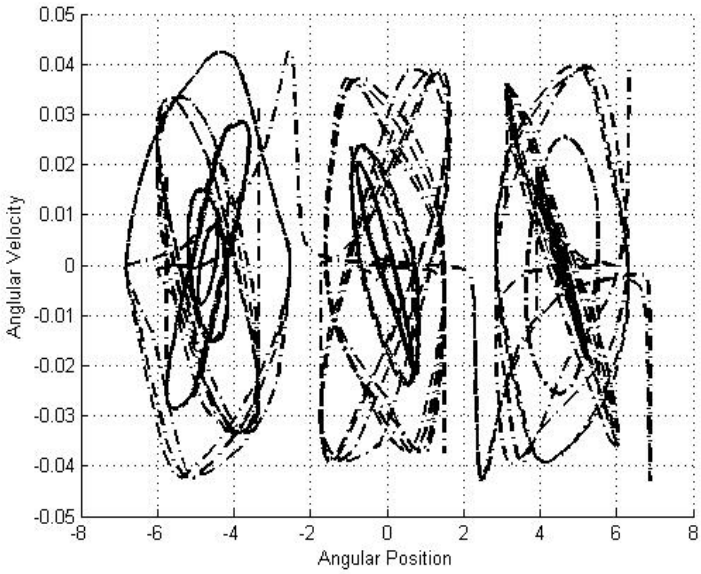


FIGURE 2 (a) The time course for the solution under various initial conditions. (b) Recurrent phase traces for the solutions shown in (a). In the range of initial conditions shown, the cycles occur at three loci, but there are potentially an infinite number, since the torque due to elasticity is a non-constant function of the initial angle. The solutions are aperiodic since the instantaneous torque increases with the angle.



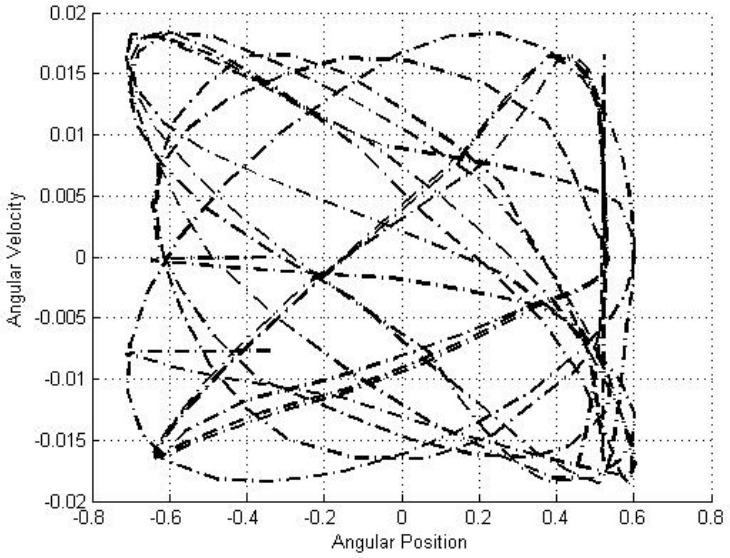


FIGURE 3 (a) We supposed that isolated cycles were obtained only when the initial position was an integral factor of 2π. Thus, we chose an aberrant step-size for the initial position, to see whether the solution would sink towards one cycle (i.e. whether it would behave as an attractor). Indeed, there was trace that connected these three cycles and seemed to fall along the trajectory of the outer two. (b) The phase response for varying initial velocities. Although the differences between the velocities are on the order of magnitude of the simulation in FIGURE 1 (a-d), we see that the system not robust.

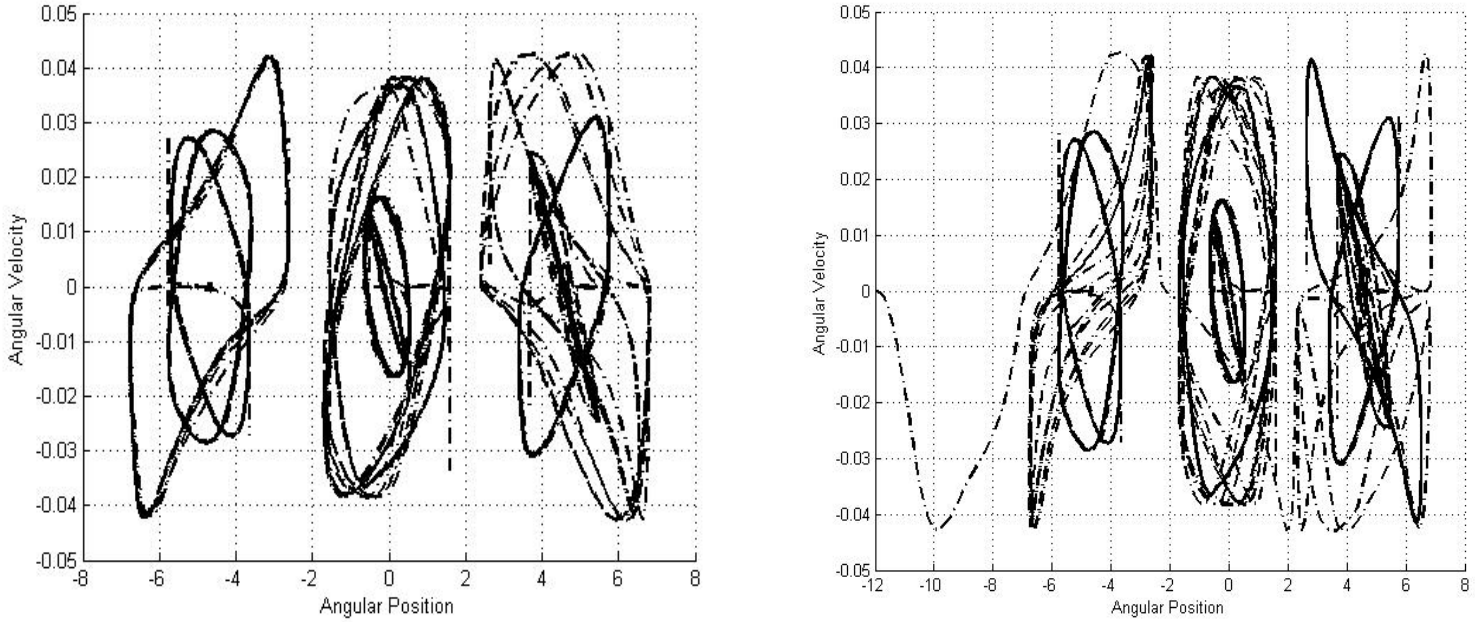


FIGURE 4 (a) Phase plane for $E_\gamma = 9 \cdot 10^{13}$. The cycles are different then those seen in FIGURE 2 (b,c). (b) Phase plane for $E_\gamma = 3 \cdot 10^{17}$. This parameter is far beyond the upper limit proposed.

5 Conclusion

Herein, we have reproduced the work of Y. Nartissov and E. Mashkovtesa. We obtained commensurate time-domain responses for the angular position and the torque due to the stalk. There is one anomaly that we were remissive about. When we simulated the model, we obtained a solution that had an opposing time course to that which was reported. Instead of a convex solution, as in FIGURE 1 (a), we found that the velocity was monotonically increasing. This is contradictory to the model, since the stalk and viscosity will always generate a moment that causes the rotation to slow. We suppose that there is an artifact in the method of the ODE solver, although only one state (the position) was discordant. Thus, we had to flip the position for each interation, and add an offset to compensate for the angle gained.

In simulating the rotation of ATP synthase, we transformed the system into state space. Since the toque due to angular acceleration was simply a linear combination of three other toques, this was facile. We had to use numerical methods to solve **(3)**. The authors do exhibit an analytic solution when only electrostatic torque is present.

We found a surprising result when we interrogated the robustness of the system. There seem to be an infinite number of periodic solutions, and these are located regularly in the phase space. In the course of discovering this, we had to assume the only active force on the system was produced by the electrostatic interactions, and that the entire ring was propelled by this force acting upon one subunit.

The efficiency with the proton gradient (a capacitance) is converted to ATP is a subject that the authors speculate about. The energy supplied the work of the ATP against the various torques is always less than $4 \cdot 10^{-21}$ J. In our simulation, this value was significantly less, since we used different parameters (E. Mashkovtesa, personal communication). This energy is not sufficient to drive ATP synthesis, which requires 55 kJ/mol. Thus, it is certain that the energy is not mechanical (i.e. caused by rotation), but rather by a more direct acid-base catalysis involving the proton gradient, and the Asp residue. The model does explain the kinetics of the rotation though, since the period of motion which is contingent on the dimensions of the relevant parameters) seems to be consistent results reported in experiments in which the rotation was directly observed (after Yoshida et al, 2001).

The implications for the various limit cycles that we found are rather abstract. Under physiological conditions, the rotation of ATP synthase would occur in an equilibrium (if both the rate-determining step and various intermediates are also in equilibrium). However, we may conclude that the Young's modulus of elasticity of the stalk (which determines the recoil which retards rotation) is very important to the stability of the system. While the system is robust to changes in this parameter (it must be on the order of magnitude of the inverse of R_γ) it does cause a bifurcation, and perhaps infinitely many, as was seen in FIGURE 4. The solution is also sensitive to initial conditions, but in native ATP synthase, deviations form from $\Theta(0)=0$ are expected to be small, since other forces would act quickly to stabilize it.

6 MATLAB Programs

```
%% Reproduction of Figures
hold on; grid on
% Parameters (F0 domain)
n = 12; Xr = 2*pi/n; k2 = 1.8096e-026;

% The end time of the simulation is approx t_span = 53.1626
%   The commented code returns this t0, so that the vector may be used for
%   the next simulation
%     [t0, X] = ode45('rotor0', [0 100], [0 0])
%     X0s = min(find(abs(X(:, 1)) > Xr));

 % The first four subunits (1 ATP)

[t0, X0] = ode45('rotor0', [0 1*t0(X0s)], [0 0]);

X0s = min(find(abs(X0(:, 1)) > Xr));

[t1, X1] = ode45('rotor1', [t0(X0s) 2*t0(X0s)], [X0(X0s,2) 0]);

X1s = min(find(abs(X1(:, 1)) > Xr));

[t2, X2] = ode45('rotor2', [2*t0(X0s) 3*t0(X0s)], [X1(X1s,2) 0]);

X2s = min(find(abs(X2(:, 1)) > Xr));

[t3, X3] = ode45('rotor3', [3*t0(X0s) 4*t0(X0s)], [X2(X2s,2) 0]);
```

```
X3s = min(find(abs(X3(:, 1)) > Xr));

% For some reason, the time course opposes that which is shown in the paper
X0(:,1) = flipud(X0(:,1)); X1(:,1) = flipud(X1(:,1)); X2(:,1) =
flipud(X2(:,1)); X3(:,1) = flipud(X3(:,1));

X0(:,1) = X0(:,1)+ Xr; X1(:,1) = X1(:,1) + 2*Xr; X2(:,1) = X2(:,1) + 3*Xr;
X3(:,1) = X3(:,1) + 4*Xr;

% Time-response
figure(1); xlabel('t 1e-5'); ylabel('Anglular Position')
plot(t0, X0(:,1), 'k', 'Linewidth', 2)
plot(t1, X1(:,1), 'k', 'Linewidth', 2)
plot(t2, X2(:,1), 'k', 'Linewidth', 2)
plot(t3, X3(:,1), 'k', 'LineWidth', 2)

% Torque from Viscosity
figure(2); hold on; grid on; xlabel('t 1e-5'); ylabel('Torque due to Bilayer')
plot(t0, k2*(X0(:,2)-X0(X0s,2)), 'k', 'Linewidth', 2); plot(t1, k2*(X1(:,2)-
X1(X1s,2)), 'k','Linewidth', 2);
plot(t2, k2*(X2(:,2)-X2(X2s,2)), 'k', 'Linewidth', 2); plot(t3, k2*(X3(:,2)-
X3(X3s,2)), 'k','Linewidth', 2);

% Phase portrait of X and X';
figure(3); hold on; grid on; xlabel('Angular Position'); ylabel('Angular
Velocity')
plot([X0(:,1); X1(:,1); X2(:,1); X3(:,1)], [X0(:,2); X1(:,2); X2(:,2);
X3(:,2)], 'k-.', 'LineWidth', 2)

%% Variation of Initial Conditions
% Phase Portraits (Positon)
figure(4); hold on; grid on; xlabel('Angular Position'); ylabel('Angular
Velocity')
t_span = t0(X0s); Xi = 2*Xr;

for i = -2*pi:Xi:2*pi
    [t0i, X0i] = ode45('rotor0', [0 4*n*t_span], [i 0]);
    X0i(:,1) = flipud(X0i(:,1))
    X0i(:,1) = X0i(:,1)+Xr;
    plot(X0i(:,1), X0i(:,2), 'k-.', 'Linewidth', 1.5);
end


% Time-domain Response (Position)
figure(5); hold on; grid on;
xlabel('t 1e-5'); ylabel('Anglular Position')

for i = -2*pi:Xi:2*pi
    [t0i, X0i] = ode45('rotor0', [0 4*n*t_span], [i 0]);
    X0i(:,1) = flipud(X0i(:,1));
    X0i(:,1) = X0i(:,1) + Xr;
    plot(t0i, X0i(:,1), 'k', 'Linewidth', 1.5);
end

% Phase Portraits (Velocity)
figure(6); hold on; grid on;
xlabel('Angular Position'); ylabel('Angular Velocity')
```

```
v_span = X(X0s, 2) - X(1,2);

for i = X(1,2):v_span:X(X0s,2)
    [t0_v, X0_v] = ode15s('rotor0', [0 2*n*t_span], [0 i]);
    X0_v(:,1) = flipud(X0_v(:,1))
    X0_v(:,1) = X0_v(:,1)+Xr;
    plot(X0_v(:,1), X0_v(:,2), 'k-.', 'Linewidth', 1.5);
end

% Iffy Initial Positions
figure(7); hold on; grid on;
xlabel('Angular Position'); ylabel('Anglular Velocity')
for i = -2*pi:Xi/1.3:2*pi
    [t0i, X0i] = ode45('rotor0', [0 2*n*t_span], [i 0]);
    X0i(:,1) = flipud(X0i(:,1));
    X0i(:,1) = X0i(:,1) + Xr;
    plot(X0i(:,1), X0i(:,2), 'k-.', 'Linewidth', 1.5);
end

%% Variation of E (Gamma)
% E = 1e7, 1e10, 1e15, 1e18

%%
figure(8); hold on; grid on; xlabel('t 1e-5'); ylabel('Anglular Position')
[t0f, X0f] = ode45('rotor0', [0 4*n*t_span], [0 0]);
X0f(:,1) = flipud(X0f(:,1));
X0f(:,1) = X0f(:,1) + Xr;
plot(t0f, X0f(:,1), 'k', 'Linewidth', 1.5);


% State-space Rendering
function [states0] = rotor0(t,y0)

% Parameters (Physical Constants)

% Physical Properties of the F0 domain
r = 3e-9; m=1.48e-22; i = m/r^2; h = 8e-9; r1 = 5.2e-9; r2 = 5.8e-9;

% Positional Properties of the F0 domain
n=12; Xr = 2*pi/n; phi0 = acos((r^2+r2^2-(r+r1)^2)/(2*r*r2)); a = pi - phi0/Xr;

% Electrostatic Properties of the F0 domain
q1 = 1.6e-19; q2 = -1.6e-19;
eps0 = 8.9e-12; eps = 2e-12;

% Viscocity of the Bilayer
eta = 0.02;

% Physical Properties of the Gamma Domain
rg = 0.8e-9; lg = 6e-9;
eg = 0.9e6; sg = 0.4;

% Absorbed Parameters
k1 = i;
k2 = 4*pi*eta*r^2*h;
k3 = (pi*rg^4)/(4*lg*(1+sg));
```

```
k5 = (r/(4*pi*eps0))*(-q1*q2/(eps*r2^2));

k1; k2; k3; k5;

% For rotor1.m, ..., rotor3.m, the above code is the same.

% State-space
Z2dot0 = (1/k1)*(k2*y0(2)+k3*eg*(y0(1)+0*Xr)-k5*sin(a*y0(1)+ phi0));
Z1dot0 = y0(2);

states0 = [Z1dot0; Z2dot0];
```