

Geometric Nonlinear Control For Acrobot Stable Walking with Changing Stepsize Gait Design and Walking on Inclined Planes

Jerry (Qilong) Cheng¹

I. INTRODUCTION

This report investigates the application of Virtual Holonomic Constraints (VHC) to control an acrobot. Various methodologies for determining VHC were explored in this study. In Part 1, VHC was approximated using polynomial functions and decomposed into a set of linear equations. Solving these equations yielded the coefficients of the polynomial. However, this approach encounters limitations due to the presence of only six linear equations, which are insufficient for optimally determining coefficients for higher-order polynomial functions.

Consequently, in Part 2, an optimization method was employed to determine the coefficients. This part explored various polynomial degrees, objective functions, and walking gait leg aperture a within the optimization framework.

Part 3 and Part 4 of the study ventured into more creative implementations, involving the acrobot ascending a slope and navigating stairs, respectively. Part 3 examined the impact of altering the gait height, unlike typical flat ground walking. Inspired by the successful outcomes of the acrobot ascending a slope in Part 3, Part 4 aimed to replicate these results for stair walking. Despite the ambitious scope, time constraints limited the successful application of optimization techniques for stair walking. To demonstrate feasibility and provide a proof of concept, a specific walking gait derived from the slope ascent was adapted for Part 4.

PART 1. STABLE WALKING GAIT DESIGN FOR THE ACROBOT

A. Virtual Holonomic Constraints

The objective of this project is to design a walking gait for an acrobot, a planar bipedal robot, using polynomial Virtual Holonomic Constraints (VHCs). The walking gait should be stable and human-like, with specific constraints on the robot's joint angles. The acrobot's configuration is defined by two joint angles q_1 and q_2 . The model is

*This work was supported by Professor Manfredi Maggiore and was dedicated for his course project, ECE1658, Geometric Nonlinear Control of Robotics Systems

¹Cheng is a Master of Engineering student under the Faculty of Electrical and Computer Engineering, University of Toronto, 27 King's College Cir, Toronto, ON M5S 1A1 qilong.cheng@mail.utoronto.ca

described by the vector $q \in \mathbb{R}^2$, where $Q = \mathbb{R}^2$ represents the configuration space.

Polynomial VHC Approach: The VHC is defined as:

$$q = \sigma(\theta) = \begin{bmatrix} q_1^+ - \theta \tilde{q}_1 \\ \phi_a(\theta) \end{bmatrix}$$

where:

- $\tilde{q}_1 = q_1^+ - q_1^-$ is the difference in the joint angle q_1 between the two specific configurations.
- $\phi_a(\theta)$ is a polynomial in θ , defined as:

$$\phi_a(\theta) = a_1 + a_2\theta + \cdots + a_k\theta^{k-1}$$

- a_i are the coefficients of the polynomial, with $i = 1, \dots, k$.

Conditions for Polynomial VHC: The polynomial $\phi_a(\theta)$ must satisfy the following conditions to ensure the proper design of the walking gait:

$$\phi_a(0) = q_2 + \quad (1)$$

$$\phi_a(1) = q_2 - \quad (2)$$

$$\phi_a(0.5) = \pi \quad (3)$$

$$\phi'_a(0) = f(v_1) \quad (4)$$

$$\phi'_a(1) = v_1 \quad (5)$$

$$\phi'_a(0.5) = v_2 \quad (6)$$

where f is a function ensuring hybrid invariance, and is defined as

$$f(v_1) = -\tilde{q}_1 \frac{(-I_{21}\tilde{q}_1 + I_{22}v_1)}{-I_{11}\tilde{q}_1 + I_{12}v_1}. \quad (7)$$

Linear System Formation: $k = 6$ is chosen in this problem formulation. The conditions can be organized into a linear system for solving the coefficients \mathbf{a} , where $\mathbf{a} = \text{col}(a_1, a_2, a_3, a_4, a_5, a_6)$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0.5 & 0.25 & 0.125 & 0.0625 & 0.03125 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 1 & 0.75 & 0.5 & 0.3125 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{bmatrix} = \begin{bmatrix} q_2^+ \\ q_2^- \\ \pi \\ f(v_1) \\ v_1 \\ v_2 \end{bmatrix}$$

After running in MATLAB, we got the following parameters for $\phi(\theta)$ $\mathbf{a} = [2.8250; 0.4246; -7.3543; 30.1847; -35.9442; 13.3225]$

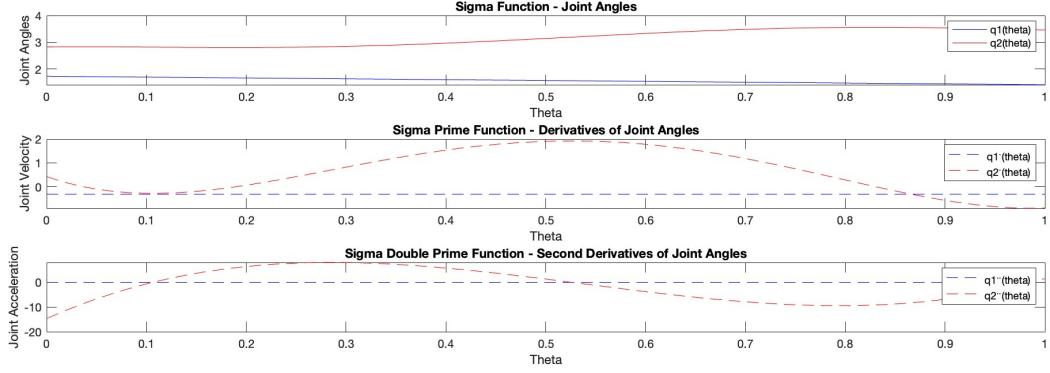


Fig. 1: θ relationship with the joint angles under the VHC constraint

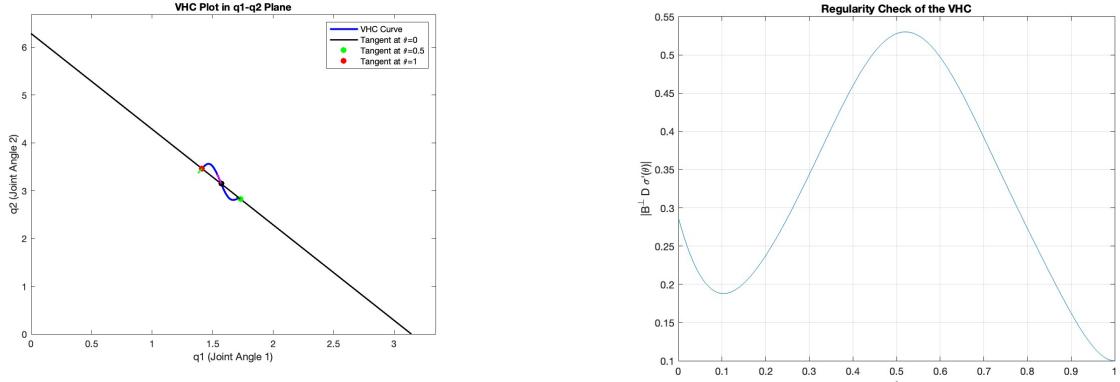


Fig. 2: VHC plot over $q_1 - q_2$. The black line is the impact surface. Blue line is the VHC curve, and the arrows indicate the tangent at the post-impact location, scuffing point, pre-impact location respectively.

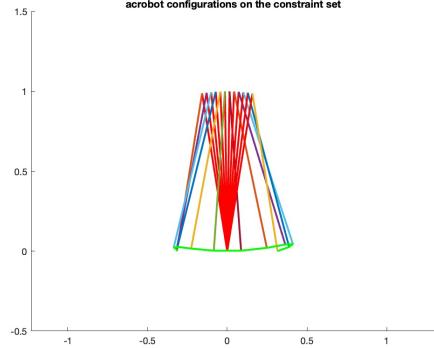


Fig. 3: The configurations of the acrobot under VHC constraint. The green line indicates the trajectory of the swing foot

B. Transversality

To verify the curve is indeed a regular VHC curve, we verified that

$$B^\perp D\sigma'(\theta)|_{q=\sigma(\theta)} \neq 0 \quad (8)$$

for all $\theta \in [0, 1]$.

Given the β , v_1 and v_2 parameters defined, we can get the following plot for the transversality:

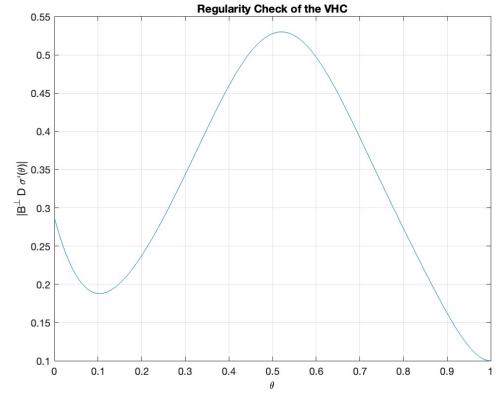


Fig. 4: Transversality condition over the range of $\theta \in [0, 1]$. Based on the figure, we can notice that there is no zero crossing, thus satisfying the transversality condition.

C. Hybrid Invariance

Hybrid invariance is crucial for ensuring that the walking gait transitions smoothly between different phases of the gait cycle. It primarily involves ensuring that certain conditions related to the dynamics of the acrobot at the points of impact are met. The slope conditions and design parameters v_1 and v_2 play key roles here.

Finally, verify whether the constraint induces a stable hybrid limit cycle. For this, we need to form expressions for ψ_1 and ψ_2

$$\psi_1 = -\frac{B^\perp \nabla_q P}{B^\perp D\sigma'} \Big|_{q=\sigma(\theta)} \quad (9)$$

$$\psi_2 = -\frac{B^\perp D\sigma'' + C\sigma'}{B^\perp D\sigma'} \Big|_{q=\sigma(\theta)} \quad (10)$$

where in this case,

$$B^\perp = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad (11)$$

The virtual mass and virtual potential energy can be computed as follows:

$$M(\theta) = \exp\left(\int_{\theta_a}^{\theta} -2\psi_2(\tau)d\tau\right) \quad (12)$$

$$V(\theta) = -\int_{\theta_a}^{\theta} \psi_1(\tau)d\tau \quad (13)$$

where $\theta_a = 0$

The resulting numerical values for M and V is plotted below against $\theta \in [0, 1]$

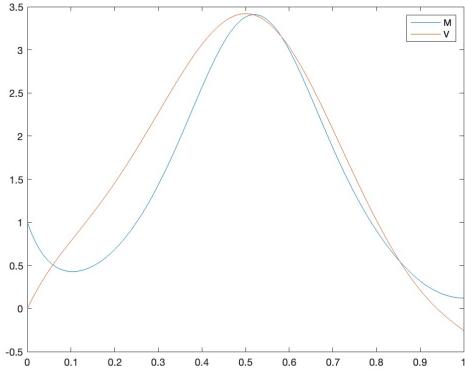


Fig. 5: The virtual mass M and virtual potential energy V over one step for the constraint dynamics

For verification, the existence and stability conditions are expressed as:

$$0 < \frac{\delta^2}{M^-} < 1 \quad (14)$$

$$\frac{V^- \delta^2}{M^- - \delta^2} + V_{max} < 0 \quad (15)$$

where $M^- = M(1)$, $V^- = V(1)$, and $V_{max} = \max_{[0,1]} V(\theta)$, and δ is given as:

$$\delta = \frac{\langle \sigma'(0), I\sigma'(1) \rangle}{\sigma'(0)^\top \sigma'(0)} \quad (16)$$

To understand the constraint dynamics better, we plotted the phase portrait of the constrained system. Each line represent the energy contour. The red line indicate the energy contour of the system when it is initialized on the limit cycle. The dotted red line represent the impact map. And based on the graph, we can clearly see that the return point satisfies the Poincarè map condition.

D. Feedback Linearization Control

Given that the error term can be expressed as:

$$h(q) = y = q_2 - \phi_a((q_1^+ - q_1)/\tilde{q}_1) = 0 \quad (17)$$

Based on this the objective of feedback linearization is to compute a control input τ that results in a desired dynamic behavior, typically forcing the system to track $\phi(\theta)$ accurately. The control law can be defined as:

$$\tau = (HD^{-1}B)^{-1} (HD^{-1}(C\dot{q} + G) - K_p \sin(y) - K_d H\dot{q} - \dot{H}\dot{q}) \quad (18)$$

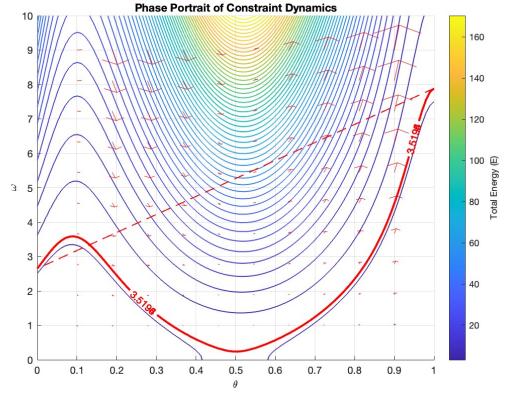


Fig. 6: Phase plot of the hybrid limit cycle when the system satisfies the Poincarè map condition. Red line indicate the energy contour at this specific configuration at the hybrid limit cycle. Red dotted line indicate the effect of the impact map.

where:

- $y = q_2 - \phi(\theta)$ is the output tracking error.
- H is the Jacobian matrix of y with respect to q .
- \dot{H} is the time derivative of the Jacobian.
- K_p and K_d scalars in this case representing proportional and derivative gains, respectively.

E. Simulation

Initial Conditions: The initial conditions are set on the constraint manifold:

$$(q(0), \dot{q}(0)) = (\sigma(\theta_a), \sigma'(\theta_a)\dot{\theta}_a) \quad (19)$$

where θ_a is an initial parameter value and $\dot{\theta}_a$ is computed from the stability analysis of the limit cycle:

$$\dot{\theta}_a = \delta_r \sqrt{\frac{-2V^-}{M^- - \delta^2}} \quad (20)$$

Here, δ_r , V^- , M^- , and δ are parameters related to the energy, mass, and damping properties of the system.

State Update: The state update, given the control input and the model for the acrobot, is:

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ D^{-1}(B\tau - C\dot{q} - G) \end{bmatrix} \quad (21)$$

F. Disturbances

During experiment, the initial conditions are perturbed to check the robustness of the system.

1) *On the limit cycle:* First, we perturbed the value, while mantaining the initial condition still on the limit cycle. In the experiment, we noticed that as long as the initial conditions are set correctly and within hybrid limit cycle, the system will remain stable and produce the same phase portrait. The only difference is the starting point of the phase portrait will be between $\theta \in [0, 1]$ and instead of $\theta = 0$.

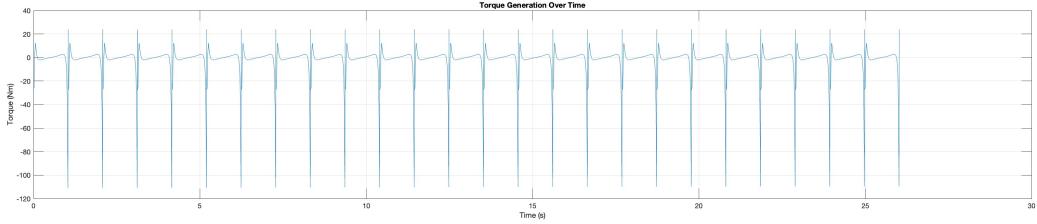


Fig. 7: Torque generated over the 25 steps simulated when the initial condition is set on the limit cycle. We can notice that the torque produced induces a stable cycle over time.

2) *On the constraint manifold:* Next, we perturbed the initial condition to be on the constraint manifold but off the limit cycle. To keep the experiment simple, we fixed the initial q position to be the same as $\sigma(0)$, while changing the \dot{q} values. In the figure below, the initial condition is perturbed with a 6% difference from the zero dynamics manifold. We can clearly see the convergence to a stable hybrid limit cycle.

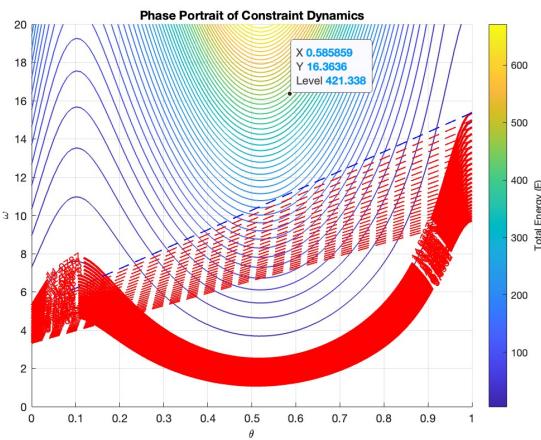


Fig. 8: Phase portrait for the constraint dynamics. The blue dotted line illustrates the first impact, and the red dotted lines illustrate the later impacts. The trend shows convergence of the system to the stable hybrid limit cycle.

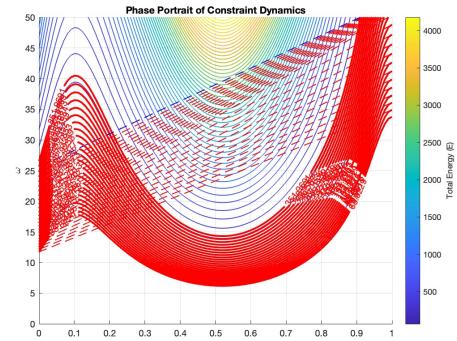


Fig. 11: Phase plot when the initial condition \dot{q} is 10 times larger than that of the limit cycle

3) *Outside the constraint manifold:* To set the initial condition outside the constraint manifold, we introduced a 10% disturbance, making the initial condition $\delta \times \sigma(0)$. During experiments, we observed that deviations from the manifold increase system fragility and reduce the likelihood of stable walking, particularly when lacking sufficient momentum for initiating movement. Nevertheless, the controller effectively managed errors within approximately 10% of the initial disturbance in q_0 . Below are results given the described disturbance.

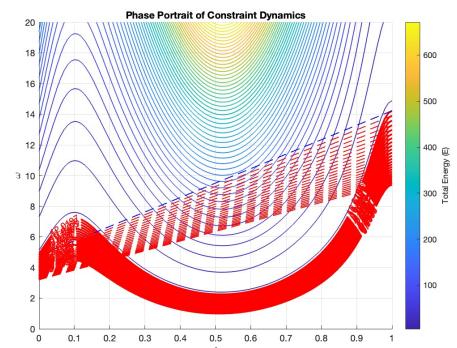


Fig. 12: Phase plot when the initial condition is outside the constraint manifold. In this case, the initial condition is set as $q = 1.1 \times \sigma(0)$

Our observations indicate that the system's tolerance to disturbances varies with the acrobot's position. For instance, at $q = \sigma(0.5)$, it can manage disturbances exceeding 1000% resulting from velocity changes, as demonstrated by the phase portrait. Despite the increased total energy from a higher initial angular velocity, the hybrid limit cycle consistently converges over time.

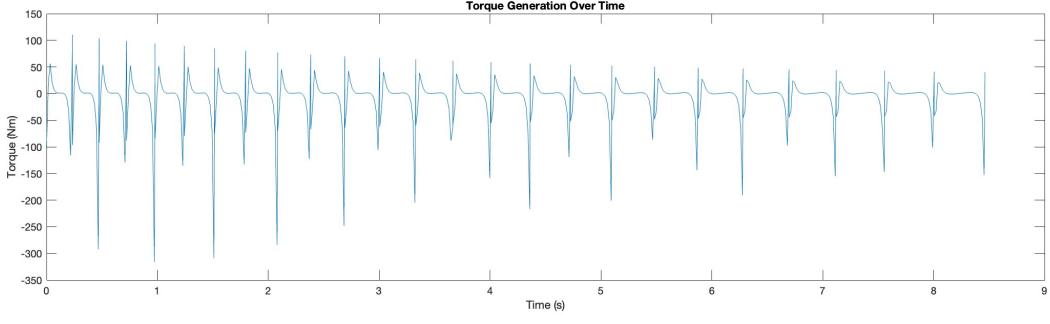


Fig. 9: Torque generated over the 25 steps simulated when the initial condition is set not on the limit cycle. the torque generated at each step assemble the similar look but varies in value. Compared to the one on the limit cycle, the torque generated here never produce the same periodic shapes.

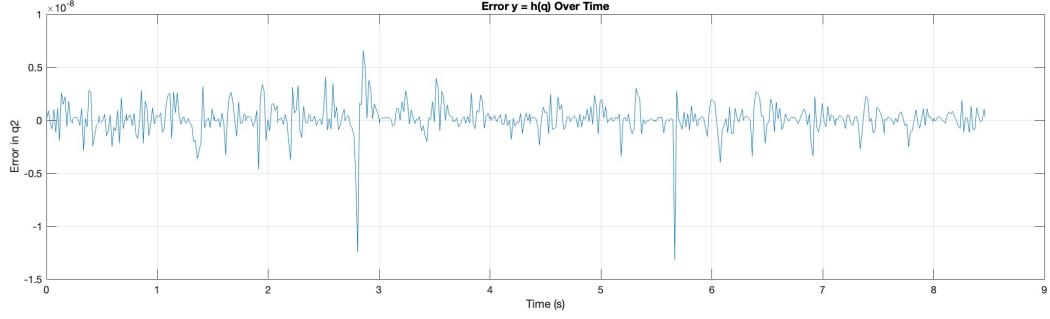


Fig. 10: Error plot when the initial condition is on the constraint manifold but not on the limit cycle. In this case, the initial condition \dot{q} is 2 times larger than that of the limit cycle

PART 2: OPTIMIZATION

G. Problem Setup

For the optimization setup, the problem is first defined as follows. We first defined the cost function J as the function we want to minimize (different objective functions have been explored in the later sections). The parameters we want to optimize is defined as $X = \text{col}(\mathbf{a}, v_1, v_2)$. Those parameters are subjected to both equality and inequality constraints defined as follow. The design parameters for the optimization includes J, k, β , representing the objective function, degree of the polynomial, and leg aperture angle respectively.

$$\text{Minimize } J(X) \quad (22)$$

Subject to:

Equality Constraints

$$\begin{aligned}\phi(0) &= q_2^+ \\ \phi(1) &= q_2^- \\ \phi(0.5) &= \pi \\ \phi'(0) &= f(v_1) \\ \phi'(1) &= v_1 \\ \phi'(0.5) &= v_2\end{aligned}$$

Inequality Constraints

$$v_1 - 2\tilde{q}_1 < 0, \quad (\text{Constraint on } v_1)$$

$$f(v_1) - 2\tilde{q}_1 < 0, \quad (\text{Function of } v_1)$$

$$2\tilde{q}_1 - v_2 < 0, \quad (v_2 \text{ relative to } \tilde{q}_1)$$

$$\zeta - (B^\perp D\sigma'(\theta_i)|_{q=\sigma(\theta_i)})^2 < 0, \quad (\text{Transversality condition})$$

$$\left(\frac{\delta^2}{M^-}\right) - 0.5 < 0, \quad (\text{Existence condition})$$

$$\frac{V^- \delta^2}{M^- - \delta^2} + V_{\max} < 0, \quad (\text{Stability condition})$$

Note: The optimization parameters $X := \text{col}(\mathbf{a}, \beta, v_1, v_2)$ have been modified to $X := \text{col}(\mathbf{a}, v_1, v_2)$ with one less variable. This adjustment is made because in the optimization process, as the optimizer aims to minimize the length of the curve, β tends to approach zero. In real-life applications, β proves to be of little use and is thus set as a design parameter. Additionally, the impact of β on the VHC constraints has been explored separately.

H. Different Degree k

Finally, we explored the impact on the number of variables to be optimized. Namely the degree of the polynomial function. In the below graph, we experimented under $\beta = \frac{\pi}{8}$ with the polynomial degree of 6, 7, and 8. The results can be seen as follows:

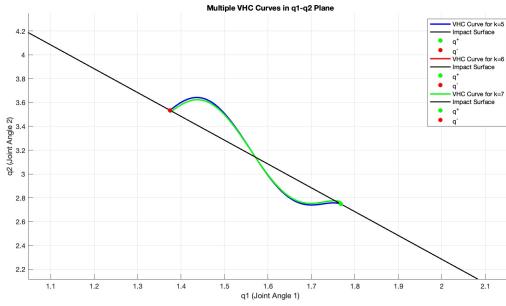


Fig. 16: Different VHC generated via the optimization approach based on different walking gait angle β

k	Coefficients	v1	v2
k = 6	13.2170, -36.2205, 30.5388 -7.2042, 0.4543, 2.7489	-1.1347	2.1743
k = 7	-0.0164 , 13.2406, -36.0385 30.2307, -7.0731, 0.4039, 2.7680	-1.2116	2.1192
k = 8	0.0038, -0.0205 , 13.2386 -36.0392, 30.2355, -7.0748, 0.4039, 2.7679	-1.2113	2.1198

Analysis of polynomial degree variations revealed subtle shifts in the Virtual Holonomic Constraints (VHC) curve, providing crucial insights for tuning the optimization solver. Importantly, coefficients at higher polynomial degrees tended towards zero, mirroring those from lower degrees, indicative of redundancy beyond the sixth degree due to only six equality constraints. This observation informed the strategy for initial solver guesses, proving that non-zero coefficients should be limited to six, with others maintained near zero to meet inequality constraints.

This discovery was instrumental in configuring initial conditions for the optimization solver, particularly in the experiments involving variations in gait step-size β . Starting with a baseline $\beta = 0.317716$, incremental increases from $\frac{\pi}{8}$ to $\frac{\pi}{3}$ were tested, with each step utilizing the outcomes of the preceding as the initial guess for subsequent optimization efforts.

This iterative approach significantly enhanced the probability of convergence in optimization, facilitating the development of diverse walking gaits later on.

I. Different Cost Functions

To compare the effects of different cost functions, we fixed the other parameters the same and compare the effect by solely changing the cost functions. The number of coefficients is set as 6; $\beta = 0.316617$.

First, we set the objective to minimize the length of the VHC curve, given as

$$\text{Minimize } \int_{\theta_a}^{\theta_b} \|\sigma'(\theta)d\theta\| \quad (23)$$

Next, we explored minimizing the square of the v_1 velocity, denoted as

$$\text{Minimize } J_2(X) = v_1^2 \quad (24)$$

Finally, we combined the effect of the two with two additional hyper-parameters γ_1 and γ_2 . The final cost function is denoted as:

$$\text{Minimize } J_3(X) = \gamma_1 J_1(X) + \gamma_2 J_2(X) \quad (25)$$

Those newly generated VHC curves are compared to when the objective function is set as the constant $J = 0$.

Experimental Results: The results can be seen as follows:

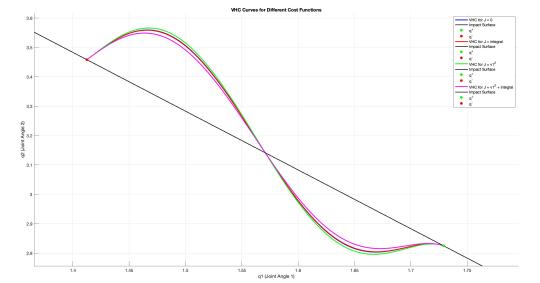


Fig. 17: Different VHC curves generated via different objective functions using the optimization approach

Cost Function	Coefficients	v1	v2
$J = 0$	13.3203, -35.9387, 30.1799, -7.3526 0.4244, 2.8250	-0.8946	1.9000
$J = \int_{\theta_a}^{\theta_b} \ \sigma'(\theta)d\theta\ ^2$	13.0847, -35.3482 29.6902, -7.2097 0.4163, 2.8250	-0.9019	1.8891
$J = v_1^2$	14.2662, -38.3018 32.0591, -7.8097 0.4194, 2.8250	-0.8987	1.9614
$J = \gamma_1 \times \int_{\theta_a}^{\theta_b} \ \sigma'(\theta)d\theta\ ^2 + \gamma_2 \times v_1^2$	11.5605, -31.5379 26.6444, -6.4512 0.4175, 2.8250	-0.9009	1.7933

From the figure, we can draw the following conclusions:

- 1) Having the objective as minimizing the curve length can induce a flatter curve.
- 2) Having the objective as minimizing v_1^2 can make the tangent of the curve at θ_a be as flat as possible
- 3) Different objective can produce slightly different walking gait depending on the requirements.

J. Different β

Since the leg aperture angle β is set as the design parameter. In the experiment, we explored different walking gait that can produce different step-size using the optimization approach. To simplify the optimization, the objective function in this case is set as $J = 0$ and the degree

for coefficient is 6. Four different walking gaits have been explored in this experiment:

- $\beta = \frac{\pi}{8}$
- $\beta = \frac{\pi}{6}$
- $\beta = \frac{\pi}{4}$
- $\beta = \frac{\pi}{3}$

The VHC curve results can be seen in the graph as follows:

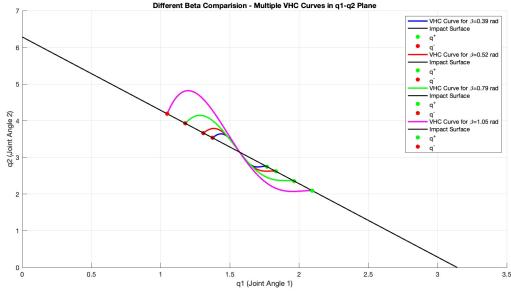


Fig. 18: Different VHC generated via the optimization approach based on different walking gait angle β

Beta	Coefficients	v1	v2
$\beta = \frac{\pi}{8}$	13.2170 -36.2205 30.5388 -7.2042 0.4543 2.7489	-1.1347	2.1743
$\beta = \frac{\pi}{6}$	12.6689 -35.8502 30.4934 -6.7565 0.4916 2.6180	-1.5973	2.6390
$\beta = \frac{\pi}{4}$	11.3745 -36.0171 31.9117 -6.1648 0.4665 2.3562	-3.3239	3.7814
$\beta = \frac{\pi}{3}$	5.8164 -34.0263 36.9993 -6.8587 0.1637 2.0944	-9.5789	5.8589

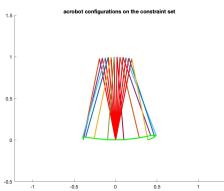


Fig. 19: Trajectory when $\beta = \frac{\pi}{8}$.

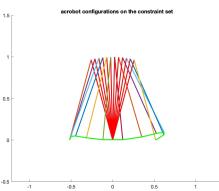


Fig. 20: Trajectory when $\beta = \frac{\pi}{6}$.

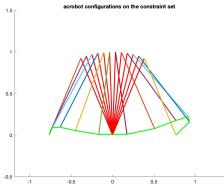


Fig. 21: Trajectory when $\beta = \frac{\pi}{4}$.

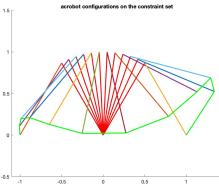


Fig. 22: Trajectory when $\beta = \frac{\pi}{3}$.

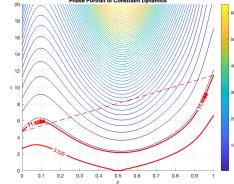


Fig. 23: Phase plot when $\beta = \frac{\pi}{8}$.

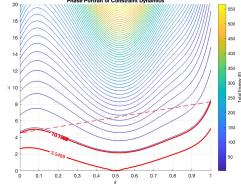


Fig. 24: Phase plot when $\beta = \frac{\pi}{6}$.

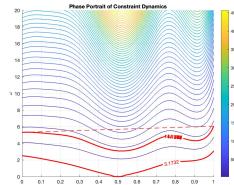


Fig. 25: Phase plot when $\beta = \frac{\pi}{4}$.

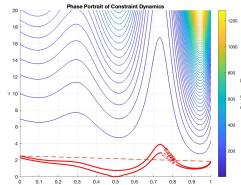


Fig. 26: Phase plot when $\beta = \frac{\pi}{3}$.

Experimental results yielded several key insights into the effects of varying the angle β on the dynamics of Virtual Holonomic Constraints (VHC):

- 1) With an increase in β , both v_1 and v_2 exhibit an increase in magnitude, suggesting enhanced dynamic responses.
- 2) The VHC curve elongates, moving the endpoints $\sigma(\theta_a)$ and $\sigma(\theta_b)$ —representing the pre-impact and post-impact points—further apart. This observation aligns with expectations, as larger β values naturally expand the distance between these endpoints.
- 3) A notable dynamic behavior change occurs with β : as β initially increases, the walking gait speed accelerates. However, beyond a certain threshold, further increases in β result in a decrease in walking gait speed.
- 4) Phase portraits reveal more complex dynamics with increasing β , characterized by curlier energy contours and multiple local minima.

Finally, in the experiment, we pushed the optimizer to optimize a VHC curve given $\beta = \frac{\pi}{2} = 90^\circ$. This experiment did not work out for me, and the optimizer cannot find a proper solution. I suspect that due to the larger step size, the required initial velocity might surpass the maximum velocity after the joint velocities are adjusted by the impact map, thus causing instability in the system.

Future Work: Investigating the maximum leg aperture β that the system can sustain without loss of stability presents an intriguing direction for further research.

PART 3. CREATIVE PART - INCLINED PLANE

When the plane is inclined, there are few things we need to modify so the existing controller can handle the inclined situation.

First, in the modeling part, the potential gradient term will take the gradient at the direction of the inclined plane direction. The vector of the inclined potential field can be expressed as:

$$\vec{g} = \begin{bmatrix} \sin\alpha \\ -\cos\alpha \end{bmatrix}, \quad (26)$$

where α is the inclined plane angle.

The new potential energy can be expressed as the previous potential energy times the potential vector.

The transformation of a gravitational vector using a rotation matrix defined by the angle ψ is given by:

$$\vec{g} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

The potential energy due to gravitational forces acting on a two-part system, assuming masses at distances l and l_c from the pivot, at angles q_1 and $q_1 + q_2$ respectively, is computed as:

$$P = - \left(mgl_c \begin{bmatrix} \cos(q_1) \\ \sin(q_1) \end{bmatrix} + mg \begin{bmatrix} l \cos(q_1) + l_c \cos(q_1 + q_2) \\ l \sin(q_1) + l_c \sin(q_1 + q_2) \end{bmatrix} \right)^T \vec{g}$$

Next, the impact map Δ will be modified too. In the relabelling map, considering the effect from α , T should be expressed as:

$$q^+ = T(q^0) = -q^0 + \begin{bmatrix} \pi + 2\alpha \\ 2\pi \end{bmatrix}, \quad (27)$$

Finally, we modify the equality conditions for σ . The new equality conditions becomes

Subject to:

Equality Constraints

$$q_1^+ = \frac{\pi + \beta}{2} + \alpha$$

$$q_1^- = \frac{\pi - \beta}{2} + \alpha$$

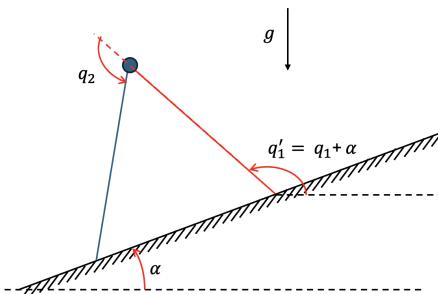


Fig. 27: Illustration for the inclined plane

Finally, during simulation, the ground impact event function is also modified. Ground impact event will occur when $q_2 + 2q_1 - 2\pi - 2\alpha = 0$ instead.

During experiment, we fixed the leg aperture $\beta = \frac{\pi}{4}$, and the degree of the coefficient as $k = 6$. The inclined angles α were chosen to be 5° , 10° , and 30° . The results are shown as follow:

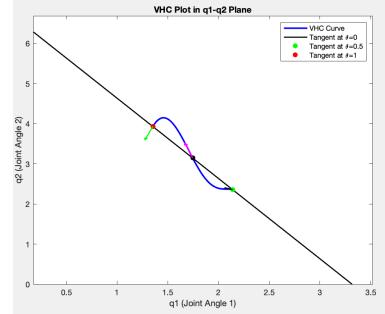


Fig. 28: VHC curve for the inclined plane given $\alpha = 10^\circ$. Compared to the flat ground, the impact surface, denoted as black line, shifted to the left with an angle α

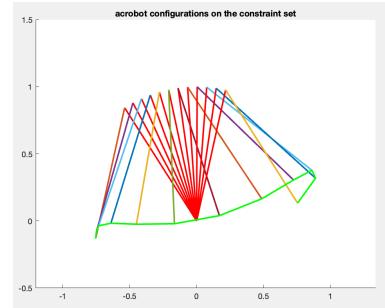


Fig. 29: Trajectory for the swing leg for the inclined plane given $\alpha = 10^\circ$. Compared to the flat ground, the trajectory tilted α angle

PART 4. CREATIVE PART - WALKING UP STAIRS

Inspired by the walking gait of walking up a inclined plane, I took the walking gait a step beyond and came up with the optimization constraints for the acrobot to walking up evenly distributed stairs.

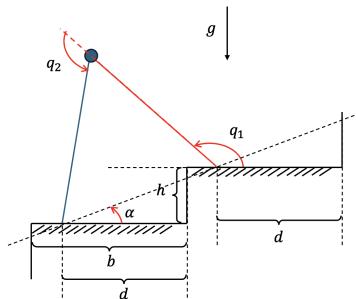


Fig. 30: Illustration for the inclined plane

The challenge of robotic stair climbing is akin to ascending a slope due to the presence of an inclined angle, denoted as α . Properly formulate this problem requires three additional parameters related to the geometry of the stairs, which are also illustrated in the accompanying figure:

- b : the width of each stair,
- h : the height of each stair,
- d : the distance from the stance foot to the bottom of the stair riser.

These parameters are essential for setting up the problem accurately. The optimization framework for navigating stairs incorporates several additional equality and inequality constraints. These constraints are crucial to prevent the robot from tripping at the stair's edge and to ensure the repeatability of each step.

Constraint derivation begins by examining the initial phase of the walking gait, where the stance foot is positioned at the rear and the swing foot is at the front.

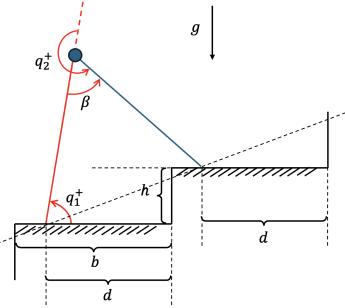


Fig. 31: Illustration for the beginning of the walking gait

After relabeling, the two feet will swap, with the stance foot at the front, and swing foot at the back.

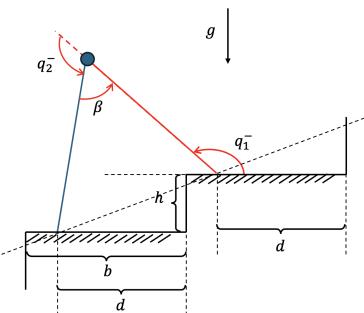


Fig. 32: Illustration for the relabeling

The equality constraints can be written as:

$$q^+ = \begin{bmatrix} \frac{\pi - \beta}{2} + \tan^{-1}(\frac{h}{b}) \\ \pi + \beta \end{bmatrix} \quad (28)$$

$$q^- = \begin{bmatrix} \frac{\pi + \beta}{2} + \tan^{-1}(\frac{h}{b}) \\ \pi - \beta \end{bmatrix} \quad (29)$$

where β in this case is determined by the stair parameters b and h , and can be expressed as:

$$\beta = \cos^{-1}\left(\frac{l^2 + l^2 - (d^2 + h^2)}{2l^2}\right) \quad (30)$$

Furthermore, the walking gait needs to satisfy the requirement not hitting the tip of the stairs. Two inequality constraints can be derived from this:

1. if $\pi - \beta_1 \leq q_2 \leq \pi + \beta_2$:

$$l \sin(q_1) + l \sin(q_1 + q_2) \geq 0$$

$$l \cos(q_1) + l \cos(q_1 + q_2) \leq (b - d)$$

2. if $\pi + \beta_1 \leq q_2 \leq 2\pi$:

$$l \sin(q_1) + l \sin(q_1 + q_2) \geq h$$

$$l \cos(q_1) + l \cos(q_1 + q_2) \geq d$$

where β_1 and β_2 are derived based on the kinematic constraints. In the first case, we need to prevent the swing foot from hitting the current stair, and we denote the angle between the two legs when the swing foot is on the boundary as β_1 .

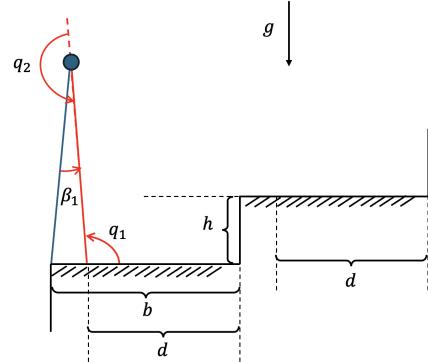


Fig. 33: Illustration for β_1 calculation

Similarly, we need to prevent the swing foot from hitting the next stair, and we denote the angle between the two legs when the swing foot is on the boundary of the next stair as β_2 .

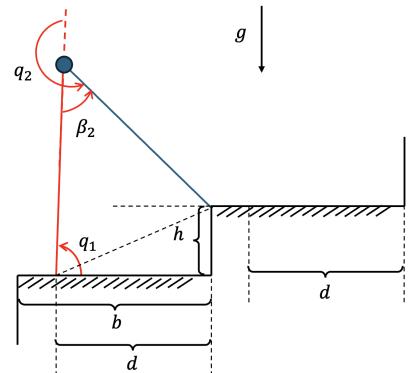


Fig. 34: Illustration for β_2 calculation

$$\beta_1 = \cos^{-1}\left(\frac{l^2 + l^2 - (b-d)^2}{2l^2}\right) \quad (31)$$

$$\beta_2 = \cos^{-1}\left(\frac{l^2 + l^2 - (d^2 + h^2)^2}{2l^2}\right) \quad (32)$$

To sum up, the revised optimization problem can be written as follows:

$$\text{Minimize } J(X) \quad (33)$$

Subject to:

Equality Constraints

$$\phi(0) = q_2^+$$

$$\phi(1) = q_2^-$$

$$\phi'(0) = f(v_1)$$

$$\phi'(1) = v_1$$

$$\phi'(0.5) = v_2$$

Inequality Constraints

$$l \sin(q_1) + l \sin(q_1 + q_2) \geq 0 \quad q_2 \in (\pi - \beta_1, \pi + \beta_2)$$

$$(d-b) \leq l \cos(q_1) + l \cos(q_1 + q_2) \leq d \quad q_2 \in (\pi - \beta_1, \pi + \beta_2)$$

$$l \sin(q_1) + l \sin(q_1 + q_2) \geq h \quad q_2 \in (\pi + \beta_1, 2\pi)$$

$$l \cos(q_1) + l \cos(q_1 + q_2) \geq d \quad q_2 \in (\pi + \beta_1, 2\pi)$$

$$\zeta - (B^\perp D\sigma'(\theta_i) \Big|_{q=\sigma(\theta_i)})^2 < 0, \quad (\text{Transversality})$$

$$\left(\frac{\delta^2}{M^-}\right) - 0.5 < 0, \quad (\text{Existence})$$

$$\frac{V^- \delta^2}{M^- - \delta^2} + V_{\max} < 0, \quad (\text{Stability})$$

Future work: This section is current still under development. But we illustrated a proof of concept in our experiment where the acrobot successfully walked up a staircase with the parameters of $b = 1$, $h = 0.5$, and $d = 0.9$. The walking gait was not solved using the above illustrated optimization setup, but derived from the optimization of walking up a slope.

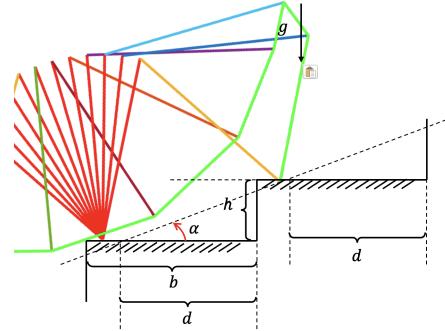


Fig. 35: Illustration for the stair walking gait

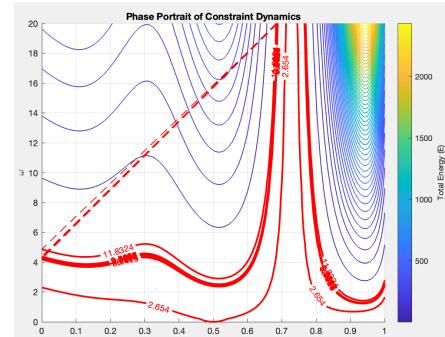


Fig. 36: Phase portrait for the stair walking gait

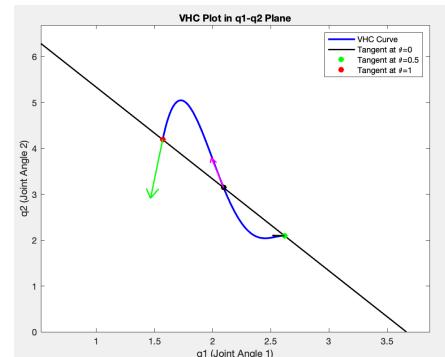


Fig. 37: VHC curve for the stair walking gait