The winch-bot: A cable-suspended, under-actuated robot utilizing parametric self-excitation

Citation

As Published
http://dx.doi.org/10.1109/ROBOT.2009.5152378

Publisher
Institute of Electrical and Electronics Engineers

Version
Final published version

Accessed
Wed Mar 13 02:24:44 EDT 2019

Citable Link
http://hdl.handle.net/1721.1/53726

Terms of Use
Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.

Detailed Terms

Daniel Cunningham and H. Harry Asada, Member, IEEE

Abstract—A simple, compact, yet powerful robotic winch, called “Winch-Bot,” is presented in this paper. The Winch-Bot is an underactuated robot having only one controllable axis. Although hanging a load with merely one cable, it is capable of moving it in a large workspace by swinging the load dynamically based on parametric self-excitation. The generated trajectories can be used for a variety of tasks, from moving material to cyclic inspection of surfaces. The basic principle and design concept of the Winch-Bot are first described, followed by dynamic modeling and analysis. Two trajectory generation problems are solved. One is point-to-point transfer of a load, and the other is the tracking of a continuous path. It will be shown that the system can track a given geometric trajectory, although the tracking velocity cannot be determined arbitrarily due to the underactuated nature of dynamics. A prototype Winch-Bot is designed and built, and point-to-point, continuous path, and parametric excitation control are implemented.

I. INTRODUCTION

There is an increasing need for a simple, compact, and flexible solution to transportation of material and end-effector in a large workspace. These needs include the manufacturing of aircraft and ships, construction of buildings and infrastructures, monitoring and inspection over broad areas, and security checks of large structures. Traditional overhead cranes are slow and bulky, and need a costly infrastructure to install and use the machine. Long arm manipulators are inefficient for covering a large space, since the actuators and arm structure become significantly larger as the required workspace increases. Mobile robots, on the other hand, are advantageous in terms of workspace to cover, but complex environments having many obstacles and rough terrain often make it difficult to deploy mobile robots.

The objective of this paper is to fill the technological gap between mobile robotics and the fixed crane and manipulator technologies. A simple, compact, winch-like robot called “Winch-Bot” will be developed to transport a heavy end-effector swiftly in a large workspace. The Winch-Bot is fixed to a ceiling or any structure above the workspace, hangs an end-effector at the tip of a cable, and swings the end-effector by parametric self-excitation. With only one or two axes of servoed joints, the Winch-Bot can control the end-effector trajectory in a large workspace.

In the robotics literature, multi-cable cranes have been studied extensively [1-3]. These advanced cranes can transport heavy objects by coordinating multiple winches that control the lengths of the multiple cables. Despite the dexterity and stability, the multi-cable cranes need multiple winch systems placed at different locations in the space. The usable workspace is substantially smaller than that of the envelope covered by the multiple winches. The casting manipulator [4], on the other hand, can project an end-effector in a long range by swinging a fly-fishing-rod-type manipulator and retrieve it by tugging the string [4]. For light payload throwing tasks, the casting manipulator approach is effective particularly for long-range projection.

The Winch-Bot consists of only one axis of winch placed at a fixed point over a large workspace. It does not need a manipulator arm to swing the end-effector; rather, it oscillates the end-effector suspended with a cable by simply extending and reducing the cable length. This is a type of parametric self-excitation governed by a Mathieu Equation. The simple, compact structure of the Winch-Bot composed of only one smart winch is advantageous for dealing with a heavy payload. Furthermore, the Winch-Bot can generate both point-to-point motion with soft-landing capabilities and continuous path motion for tracking a geometric trajectory. This allows us not only to throw an object but also to track along a surface, search a wide area, and perform a variety of tasks that require path control capabilities.

Fig. 1. Shown are the parameters important in our simple pendulum whose length is controlled by a smart winch.

In the following, the basic principle and design concept of the Winch-Bot under-actuated system will be presented, followed by dynamic modeling and trajectory generation algorithms. A prototype Winch-Bot and its initial experiments will be described to evaluate the feasibility and potentials of the approach.
II. PRINCIPLE

A. Parametric Self Excitation as a Thrust for Under-actuated Robots

Consider a two-dimensional pendulum system within a vertical plane. The system consists of a single-axis winch fixed to a ceiling or any fixed point above a workspace, shown in Fig. 1. A cable hangs down into the workspace, and at the end of the cord is attached the end-effector, whether it be a basket, a hook, a sensor platform, or anything with a larger mass. Let \( r \) be the length of the cable, i.e. the distance between the end-effector and the fixed point \( O \) at the ceiling, and \( \theta \) be the angle of the cable measured from the vertical line. The length of the cable, \( r \), is actively controlled. No other means of actuation is needed for transporting the end-effector in the two-dimensional space. Hence the system is under-actuated.

When the cable length is fixed, the system is a simple pendulum oscillating at a specific amplitude and frequency. As the winch varies its cable length dynamically in relation to the cable angle \( \theta \), the oscillation amplitude is increased or decreased depending on the “phase” between the two time profiles for \( \theta \) and \( r \), a well-known dynamic phenomenon, “parametric self-excited vibration.” Imagine a child on a playground swing set. The child can “pump” his/her legs back and forth in time with the swinging. This way the thrust the child is using to pull his/her legs back and forth is transferred to the motion of the swing. This is the same principle of our system, where we use energy to change the length of the cable in time with the swinging motion, which generates thrust to the swing, thus increasing the amplitude of the swing. As will be discussed later, the thrust can be synthesized more deliberately than just for swinging, so that a desired end-effector trajectory can be generated.

With only a single actuator, only in-plane motion is affected through parametric self-excitation. With the addition of another actuator, out-of-plane motions can be controlled. Simple ways to achieve these out-of-plane motions include introducing a rotation axis to the fixed winch, or by placing the wince on a sliding rail. By driving the rotation or out-of-plane motion as a function of the in-plane swing motion, motion perpendicular to the swing plane can be induced. Moving the out-of plane motion will induce Coriolis forces in the end mass, so much more research into its control would be required, but is not covered here. Examples of this implementation are shown in Fig. 2.

B. Transportation Tasks

This Winch-Bot can generate a variety of controlled oscillatory movements for transporting an end-effector in a large workspace. In contrast to traditional gantry robots and long-arm manipulators, this Winch-Bot is much more compact and simple, yet applicable to a variety of tasks.

Two types of transportation tasks appropriate for the Winch-Bot are illustrated in Figs. 3 and 4. In Fig. 3, the end-effector must be moved from a specified start point in space to another specified point.

![Fig. 3. The most common endpoint control scenario is the lift, leap, and landing. Here, only the initial and final endpoints are specified.](image)

Fig. 3 also shows a typical trajectory of the end-effector leaping from initial point A to destination point B. If the initial point A is not exactly beneath the winch, point O, but at a point having some offset from it, the end-effector begins to oscillate as soon as it is lifted by the winch. Accommodating the oscillation amplitude by means of the parametric self-excitation, the Winch-Bot can accumulate energy to reach the specified final point. The trajectory may be elaborated so that the end-effector may land on the destination at zero velocity, i.e. a soft landing.

![Fig. 4. Continuous path control is useful when the tracking velocity is unspecified, but an entire trajectory within a range is desired.](image)

Fig. 4 illustrates another type of tasks: continuous path (CP) control. In the figure, the end-effector carrying an inspection instrument is moving along a wide surface, an aircraft body, for example. The task is to move the end-effector a few centimeters above the surface at an unspecified velocity. The geometric trajectory is specified and the end-effector must keep the distance from the surface. It must be noted, however, that the speed of the end-effector along the trajectory cannot be specified due to the underactuated, self-exciting nature of the system dynamics. There are many tasks where speed control is not a stringent...
requirement, but geometric trajectories are. For a class of trajectories, there exists an input pattern to the winch control system that drives the end-effector to follow the specified trajectory.

Input pattern synthesis is a central issue for this type of underactuated robots. The following sections will describe dynamic modeling and input syntheses for each type of task: point-to-point and continuous path.

III. DYNAMIC MODELING

A. Equations of Motion

The governing dynamic equations of the Winch-Bot are derived in this section. We assume that the end-effector is a point mass and that the cable is mass-less. We also ignore the longitudinal elasticity of the cable and aerodynamic effects on the end-effector and the cable, which measured to be small in comparison to the other forces and velocities.

![Diagram of a point mass connected by a cable](image)

Fig. 5. The well-known free-body diagram of a point is relevant to our setup.

Starting with the basic free-body diagram for a point mass, one can derive the two equations of motion for our fixed point/variable-length cable system, moving only in a single plane. Let \( m \) be the mass of the end-effector and \( T \) be the tension of the cable. See Fig. 5. Assuming that the cable is taut, we can obtain the following equations of motion:

\[
(mg \cos \theta - T) \dot{e}_r - (mg \sin \theta) \dot{e}_\theta = m \frac{d^2}{dt^2}(\dot{e}_r) \tag{1.1}
\]

where \( \dot{e}_r \) and \( \dot{e}_\theta \) are unit vectors pointing in the radial and tangential directions, respectively. Expanding the right-hand side and taking a time derivative, we get

\[
(mg \cos \theta - T) \dot{e}_r - (mg \sin \theta) \dot{e}_\theta = m \left[ \frac{d}{dt}(r \dot{\theta}) \right] \dot{e}_r + m \left( 2r \ddot{\theta} + r \dot{\theta}^2 \right) \dot{e}_\theta. \tag{1.2}
\]

From the \( \dot{e}_r \) direction we get

\[
g \cos \theta - \frac{T}{m} = \ddot{r} - r \dot{\theta}^2, \tag{1.3}
\]

and from the \( \dot{e}_\theta \) direction we get

\[
-g \sin \theta = 2r \ddot{\theta} + r \dot{\theta} \tag{1.4}
\]

or

\[
\ddot{\theta} + \frac{2r}{r} \dot{\theta} + \frac{g \sin \theta}{r} = 0. \tag{1.5}
\]

Equation (1.5) is our main dynamic equation to be used for control synthesis. Equation (1.3) is needed for two purposes:

- To check whether a trajectory under consideration violates the actuator torque limit or not, and
- To ensure that the tension is always positive, meaning the cable stays taut.

When the cable length \( r \) is varied periodically, the oscillatory behavior of the Winch-Bot can be described with a Mathieu Equation, whose properties have been well documented. We will use the known properties of the Mathieu Equation for trajectory planning, in particular, for accommodating the swing amplitude and energy level. However, for more general trajectory synthesis we will explore a broader class of input patterns based on the full dynamic equations, (1.3) and (1.5).

IV. POINT-TO-POINT LEAP TRAJECTORY PLANNING

One use for this type of system is that of lift, swing, and land, all in a single plane. In order to generate a valid trajectory to get the endpoint from one arbitrary place to another, we must describe the problem in such a way that a solution can be systematically found. Optimal control is a way of rigorously obtaining a valid trajectory. However, it often ends up with an impractically complex numerical method and heavy computational load. The following three-step algorithm following the lift-swing-land sequence provides us with a practical, yet efficient solution method utilizing a bidirectional planning algorithm first demonstrated by Nakamura et al [5].

**Step 1: Lift**

The first segment, the Lift, consists of the dynamic motion between an initial state and a higher-energy state with a shorter string, caused by a shortening of the cable. See Fig. 6-(a). There are numerous trajectories connecting the initial and final cable lengths. To facilitate computation, let us parameterize the string length trajectory as a sigmoid curve. As shown in Fig. 6-(b), a sigmoid trajectory generates a smooth pull from the initial point to a higher height. The change of length of the string is \( \Delta r_p \). If \( \Delta r_p \) is known, then the only parameter to be varied in this segment is the amount of time it takes to pull the cable, \( \Delta t_p \).

![Diagram of a lift segment](image)

Fig. 6. The first step in the sequence of calculated events is the pull, or where the string is shortened to clear the floor, to have a higher frequency, and to drive up the energy in the swing to a desired state.

Changing this length of time parameter has the effect of varying the resulting energy in the pendulum’s swing cycle. If one imagines pulling a string to a shorter length at a speed that finishes the pull as the pendulum passes through one half-period, the amount of resulting energy is much higher than if the pulling occurred over a full period. Fig. 7 shows the effect the time parameter has on the final energy of a system going from one initial point and pulled through a
specific $\Delta p_t$. This time parameter $\Delta t_p$ is varied to best match the energy at the beginning of the Landing segment. This technique is similar to the principle of Time Scaling used by Arai et al [6], [5].

![Chart showing change in energy per unit mass (J/kg) vs. time to pull (s) for trajectories with different pull times.](image)

**Fig. 7.** How quickly one decreases the length of the string affects the resultant energy in the eventual stable swing. This plot shows results from a pendulum with a starting length of 2.33 m and a $\Delta p$ of 0.83 m. The insert plots show the pendulum trajectory at specific points.

**STEP 2: SWING**

The second segment, or Swing, is necessary if the energy in the pendulum at the end of the Lift does not match the energy required for the final landing. This could be caused by errors in measurement, or only because the desired energy cannot be achieved with the Lift alone. In this segment the energy within the pendulum’s swing is adjusted using parametric oscillation to generate self-excitation (or damping). These parametric oscillations are calculated utilizing the general solutions to the Mathieu Equation.

![Diagram showing swinging pendulum.](image)

**Fig. 8.** The second step in the sequence is not always needed. When required, this step uses parametric self-excitation to adjust the energy within the swing.

**STEP 3: LAND**

The third and final segment is Land, which is the Lift in reverse time. It starts from the trajectory achieved by either the Lift or the Swing, and smoothly extends out to the final point, again parameterized as a sigmoid curve. In most cases, the final point has zero velocity, so it is obvious how this is basically a reverse-time version of the Lift segment; instead of going from a fixed point with no velocity to periodic pendulum motion, the system goes from a periodic pendulum motion to a fixed point with no velocity. Not surprisingly, this step is solved mathematically just as one would expect: set the final point as initial conditions, and solve from time $t = 0$ to $t = -\infty$. (In reality, going backward, the system reaches the desired energy rapidly, so it is unnecessary to solve for $t$ of large negative values.)

![Diagram showing a pendulum landing.](image)

**Fig. 9.** The third step takes the system from a stable swing cycle, lengthening the string to softly land with zero velocity at the endpoint.

In total, our strategy is to pick an arbitrary middle height (from where the Land starts the Lift ends), then to calculate the Land in reverse time. This solution gives us the energy required from the first two steps. From there we take the initial conditions, adjust the pull time so that the Lift reaches the desired middle energy. If this is impossible, the Swing step will be necessary, and controlled in real-time.

In the case of the endpoint being at rest, the initial conditions can be measured by the system itself. As long as the cable is not slack, the measured angle of the cable combined with the current length fully defines the location of the endpoint. This is most useful when the endpoint is a tool holder and can be placed in a constant, fixed, location.

Other situations may arise where the initial conditions are not at rest, such as in a stable oscillatory trajectory. If that is known, the system can use that as initial conditions, because it can measure the progress around the oscillations and trigger based on those measurements.

Specifying the final conditions includes more than just specifying a final endpoint location. If this system were used to apply force at a specific location, like for “hammering” in a series of nails or impact-testing at multiple points, one can specify a final location and a non-zero end velocity, which would translate to a force dependent on the mass.

**V. CONTINUOUS PATH CONTROL SYNTHESIS**

Continuous path control synthesis is the problem of finding an input pattern for the winch so that the end-effector can track a given geometric trajectory. Unlike standard continuous path control for fully actuated systems, we cannot specify a time trajectory for the end-effector since the Winch-Bot is an under-actuated system. When a geometric trajectory is specified, the time trajectory is to be determined from the conditions that dictate the end-effector to follow the geometric trajectory. Before solving a general CP control synthesis problem let us work out a special case, which is of practical importance.

**A. A Special Case: Horizontal Trajectories**

Consider a horizontal trajectory, as shown before in Fig. 4. The end-effector is required to track this horizontal line at a distance $h$ from the winch center. Let $s$ be the distance along
the horizontal line measured from point C, which is directly beneath the winch center. For the end-effector to move along the horizontal line, the cable length must be varied to satisfy

$$r(s) = \sqrt{s^2 + h^2}$$  \hspace{1cm} (1.6)$$
and the cable angle must be

$$\theta(s) = \tan^{-1} \left( \frac{s}{h} \right).$$  \hspace{1cm} (1.7)$$
When these conditions are perfectly satisfied, then the end-effector mass \(m\) has no acceleration in the vertical direction. Therefore the cable tension is given by

$$T = \frac{mg}{\cos \theta}$$  \hspace{1cm} (1.8)$$
and its component in the horizontal direction given by

$$F_y = -mg \tan \theta = -\frac{mg}{h} s.$$  \hspace{1cm} (1.9)$$
which works as a braking force on the end-effector as it tends to move away from point C. Therefore, the work done by the winch upon the end-effector when it moves from point C to a current position at \(s\) is given by

$$\text{Work} = \int_0^s F_y ds = \frac{mg}{2h} s^2.$$  \hspace{1cm} (1.10)$$
Suppose that the end-effector passes point C at a speed \(V_C\), where it has kinetic energy \(T = mV_C^2 / 2\). The end-effector can move along the line until the winch absorbs all the kinetic energy. Let \(s_D\) be the distance to point D where the kinetic energy becomes zero. Equating Work and \(T\) yields

$$s_D = V_C \sqrt{\frac{h}{g}}$$  \hspace{1cm} (1.11)$$
On the way back from point D, the end-effector regains the kinetic energy and continues to move the same distance \(s_D\) in the opposite direction, if the process is loss-less. Therefore, the Winch-Bot can perform the continuous pass tracking of the horizontal trajectory of length \(2s_D\). The tracking speed for the forward motion is given by

$$V(s) = \sqrt{V_C^2 - \frac{g}{h} s^2}.$$  \hspace{1cm} (1.12)$$

The true system has some friction and energy dissipative component. Therefore, the distance of horizontal motion, \(2s_D\), decreases if the same horizontal trajectory tracking must be repeated many times. To resolve this diminishing distance problem, energy must be pumped into the end-effector. As shown in Fig. 10, we can increase the energy level by using the same techniques described in the previous section to augment the trajectory.

### B. General Case

![Fig. 11. The general case for CP control makes the variable \(s\) arbitrary, and both \(\theta\) and \(r\) are functions of that variable \(s\).](image)

Based on the insight gained from the above special case, let us now formulate a CP control synthesis problem for general trajectories. To facilitate to describe a geometric trajectory, consider a path length \(s\) along a given trajectory. See Fig. 11. Given a geometric trajectory, the cable length and the cable angle must satisfy functional relationships so that the end-effector may track the trajectory:

$$r = r(s), \quad \theta = \theta(s).$$  \hspace{1cm} (1.13)$$
The tracking speed is given by

$$\dot{s} = \frac{ds}{dt}$$  \hspace{1cm} (1.14)$$
and the time derivatives of \(r\) and \(\theta\) are given by

$$\dot{r} = \frac{dr}{ds} \frac{ds}{dt} = r \dot{s}, \quad \dot{\theta} = \frac{d\theta}{ds} \frac{ds}{dt} = \theta' \dot{s}. $$  \hspace{1cm} (1.15)$$
Furthermore, the second derivative of \(\theta\) is given by

$$\ddot{\theta} = \theta'' s^2 + \theta' \ddot{s}.$$  \hspace{1cm} (1.16)$$
Substituting these into (1.5), our main dynamic equation, yields

$$\ddot{s} + A(s) \dot{s}^2 + B(s) = 0,$$  \hspace{1cm} (1.17)$$
where

$$A(s) = \frac{\theta''}{\theta'} + \frac{2r'}{r},$$  \hspace{1cm} (1.18)$$
and

$$B(s) = \frac{g \sin \theta}{r \theta'}.$$  \hspace{1cm} (1.19)$$
Given a geometric trajectory and initial conditions, (1.17) can be solved numerically for time function \(s(t)\). Once the time profile of path length \(s(t)\) is obtained, the cable length and the cable angle as well as their time derivatives can be obtained. Based on these results, we examine whether the solution is eligible. Typically we have to check the following items:

- The solution \(s(t)\) covers the entire geometric trajectory.
- The tracking speed \(s(t)\) is within an acceptable range.
The cable tension $T$ is always positive, i.e. no slack of the cable.

- The cable tension $T$ and the winch speed $\dot{r}(t)$ is within the maximum actuator parameters.

These conditions depend on the initial conditions of the end-effector. To find an acceptable solution, the above numerical procedure may need to be repeated for varied initial conditions. Note that the initial conditions can be tuned by augmenting the trajectory and applying the same techniques described previously.

VI. IMPLEMENTATION

To test the system in question, we built a pendulum with a winch-controlled string. The mass at the end was a steel ball encased in plastic, and the string was Kevlar, to reduce the elastic effects of a string. The winch was a coil of the string, driven by a DC motor. We instrumented the motor with a precision encoder that would feed back the string’s actual length in real-time. To measure the string’s angle we used a light source to cast a shadow of the string onto an optical sensor, which translated the signal to the string’s angle relative to the structure. With these two parameters we would know the state of the system within the plane of desired swing motion. We used a simple closed-loop controller to specify the desired position of the winch, which drove the length of the string to our desired length.

Fig. 12. A light-tight box encasing a single LED was needed to reduce the effects of ambient lighting on the optical sensor.

Our implementation is shown in Fig. 12. Included is a rotary motion stage and actuator for further study into the modes of swinging with this additional degree of freedom. In our primary topic, the trajectory planning algorithms, this degree of freedom was not used. Implementation of the optical angular measuring device required the addition of a light-tight box to eliminate the effect of the room’s ambient light.

VII. EXPERIMENT

In our initial experimentation, we looked at three particular segments of the trajectory generation: the parametric excitation, a point-to-point trajectory, and path tracking using angular feedback.

To parametrically excite the system, we designed a trajectory with a smooth pull like in Fig. 6-(b), and then a driving frequency related by the Mathieu solutions. The intent was to start with a small perturbation in the swing direction, and have the system parametrically drive itself to states of much higher energy. In fact, we were able to give the mass a small push, and the system would quickly drive itself to large swinging angles.

Using the technique described in section IV, we chose a start and an endpoint, and varied the pull time to match the energies of the two pulls. Then we joined the trajectories (with a proper amount of time between to allow for an integer swing period), and used the trajectory as a moving setpoint for a proportional controller. The tracking results are shown in Fig. 13. The setpoint and resulting measured length is shown in Fig. 13-(a), where the actual measured angle is presented with the simulated angle in Fig. 13-(b).

By combining the recorded data from the actual length and actual angle of the string, plotting the X-Y position is more visibly indicative of the success of the trajectory generation. Fig. 14 shows the actual X-Y trajectory against the simulated, as would be seen from a point away from the swing plane.

Fig. 14. The simulated X-Y position of the system is shown as a solid grey line, and the actual trajectory is shown as points. The actual initial pull sweeps higher than simulated due to release timing errors.
In general, the system followed the generated and simulated trajectory well. The initial pull is difficult to match, as coordinating the start of the swing with the zero-time in the generated trajectory is difficult. These errors caused the first swing to rise too quickly (because the release was after the trajectory zero), as can be seen in Fig. 14. Despite this, the final position and velocity matched the desired state nicely. This likely implies that the system is less sensitive to disturbances to or improper measurements of the initial state as previously expected.

To experiment with path tracking, we calculated a fairly smooth path trajectory that resembles the cross-sectional contour of an airfoil, shown in Fig. 15. We then created a closed-loop control system that would measure the angle of the string and calculate the string length that would place the endpoint on the prescribed curve. That calculated string length would become the new setpoint of the length controller. Fig. 15, an X-Y plot of the desired geometric path and the measured trajectories, shows that the controlled trajectory generally tracks the prescribed path, as long as the change in the setpoint is small.

Fig. 16 shows the absolute error in length between the prescribed geometric path and both a typical free-swinging pendulum and one actively tracking the path.

As long as the angle-measure commands only small step changes in the setpoint, the path is smooth. If the geometric curve is too steep or a large disturbance affects the measurements, a large step change in the setpoint causes the controller to accelerate the endpoint mass too quickly, causing the mass to “lurch.” When this occurs, because it is not a point mass, the mass begins to wobble about its own center, shaking the string. Fig. 17 shows an example of the “lurch” and the subsequent errors in string measurement.

![Graph](image)

**Fig. 15.** A simple PD controller is sufficient to generally track a prescribed endpoint trajectory.

**Fig. 16.** While a simple pendulum has large errors toward the far left portion of Fig. 15, the simple controlled pendulum reduces those errors dramatically.

**Fig. 17.** In these two trials of string-feedback path-tracking, the first (grey) tracks well, but the second (black) lurches at around 6.1 seconds, likely from an outside disturbance. This causes accelerations larger than that of gravitational acceleration, causing the mass to free-fall, thus disturbing all further string measurements and control.

**VIII. CONCLUSION**

Initial experimentation shows that the technique of using planned dynamic trajectories to increase the workspace of our small Winch-Bot succeeded and was controllable enough to specify initial and final points. Even using only a feed-forward method of control was successful in moving the end-effector to the final point. It was also observed that real-time control of the trajectories could be done by varying the driving self-excited oscillations, to change the energy. This allows us to make up for errors in measurement, modeling, and implementation in real-time, which is important for controllability of the system. Additionally, the opportunity for CP control is attractive, and we showed that it is achievable for certain classes of trajectories.

**IX. ACKNOWLEDGEMENTS**

We would like to thank the Boeing Corporation for their sponsorship of this project. Without them, there would be no drive for this technology. Also, Ian Rust was instrumental in developing and building a wireless end-effector distance-measuring device, which allowed us to more accurately track the CP trajectories.

**REFERENCES**