Accelerometer Based Self-Discovery of Multiple Link Kinematics

Jeffrey R. Croxell¹, Ryan W. Krauss², Jerry B. Weinberg³, and Scott R. Smith¹

¹ Department of Electrical and Computer Engineering, ² Department of Mechanical Engineering, ³ Department of Computer Science
Southern Illinois University Edwardsville
Campus Box 1656
Edwardsville, IL 62026
jcroxel@siue.edu

Abstract
Robot arms have gained acceptance as a valuable tool and are in wide use, largely because of inverse kinematics. Despite their popularity, few people have the technical knowledge to implement inverse kinematics on a robot arm. The use of accelerometers can significantly reduce the difficulty of implementing inverse kinematics on robotic arms by automating the setup process. A solution to find arm parameters such as link length and link twist for the purpose of solving kinematic equations is described. Considerations to improve the results gathered from an accelerometer-based approach are discussed.

Introduction
Typically robot arms are operated using inverse kinematics. Inverse kinematics allows the user to specify a point in three dimensional space and move the tip of the arm to that location. This is convenient, but can be complex if the user is not experienced in the area of robotics, kinematics, and control.

The use of robot arms as well as humanoid robots is becoming more popular amongst hobbyist as well as students of all ages and abilities (Weinberg et al. 2008). Previously, they had been a tool confined mostly to the fabrication industry as well as some research in university settings. Recently the availability of humanoid and legged robots has excited younger students (Figure 1) not formally trained in engineering and contributed to the growing industry in the toy, education, and consumer robotics market. However, these perspective robot engineers can soon find that the use of robot arms and humanoid limbs is a math intensive task, so much so that it may bar usage entirely. Instead, students utilize preprogrammed positions for each joint on an arm. Regardless of how many degrees of freedom, this is a very tedious and time consuming task, as well as very confining. The user will have only a finite set of locations defined to position the tip of the robot, not a full range that is available with inverse kinematics.

This is unfortunate, unfair, and unnecessary. Today, people make use of technology freely without a second thought of how it works, or the need to. This same concept could be applied to robot arms, opening a new set of possibilities and ease of use.

To automate the setup of inverse kinematics we will make use of an accelerometer on the tip of the robot arm. Accelerometers have become increasingly used in products from cell phones to cars and even video game systems. Their low cost and unique feedback makes them an excellent choice.

Accelerometers have been used in robotics for various forms of feedback, such as surface detection (Sinha and Bajcsy 1992) and navigation (Barshan and Durrant-Whyte 1995). The concept of automation applied to robot arms and kinematics has little or no previous work. Chandrasekar and Bernstein (Chandasekar and Bernstein 2008) have utilized accelerometer feedback in simulation for position control. (Canepa, Hollerbach, and Boelen 1994) used accelerometers to find kinematic parameters but utilized tangential acceleration in their method of length measurement.

Figure 1: Young students enjoying RoboGames. Notice the large number of joints on each arm and leg requiring positions (Ottawa Citizen 2007).
Method

This project aims to automate the process of setting up an inverse kinematic solver for the user. Since the process is automatic, clear definitions of knowns and unknowns are necessary. For this work we will always assume joints have an offset of zero.

Most robot arms have feedback from each of the joints, typically encoders; this will be one of the tools the system knows how to utilize. We assume that using these encoders we can find the zero or fully extended position of the arm for each joint. With the use of the encoders we will also inform the system how to operate the motors themselves. The final known used by the system will be the accelerometer, which is not typically found on robot arms but is the integral piece to our system to discover characteristics about the arm.

Link Measurement

To setup kinematic equations for any robot there are several physical characteristics that must be configured and computed for the specific arm in question. For proper operation, accurate measurements of the length of each link on the arm must be found.

A simple algorithm based on circular motion can give us the required length of a given link (Hibbeler 2004). As seen in Figure 2, a link is rotated at a constant velocity, which is calculated by differentiating our encoder readings at constant intervals. Using the angular velocity with the normal acceleration measured by the accelerometer gives us the length of the link:

\[ l = a_N/\dot{\theta}^2 \quad (1) \]

This equation is both simple and convenient. It is continuously calculating the length of a given link regardless of what speed the joint is moving or the amount of acceleration being sensed. As we see in Figure 2 the link is rotating at a constant angular velocity, but a longer link \( l_2 \) would have a higher velocity and larger normal acceleration at the tip. This also means with very short links at low velocities measurements may become less accurate.

This equation works well for one link with one accelerometer but we then need to extend this to arms with multiple degrees of freedom (DOF). To accomplish this, a methodical set of movements in a calibration routine are necessary. Figure 3 shows an example of what is needed to calculate all links in a 2 DOF arm. All joints begin in a static position, with the first joint moving in Step 1. Using equation (1) we find \( l_1 \)’s length. At this first step the joint number and link are ambiguous to the system. The arm then returns to the zero position and begins moving another joint (Step 2) to find the next length \( l_2 \). Now that a second joint and link have moved the position of each \( l_1 \) and \( l_2 \) are clear, as well as the order of the joints. We can deduce that one of these measurements was the overall length and one was an individual link length. This can be used to find the length of the individual link lengths as seen in Step 2 of Figure 3.

Joint Orientation

To build an internal kinematic model, we must also know at what angle a joint meets another link, this is known as link twist. This would also indicate what plane that portion of the arm operates in, and eventually lead us to the arm’s workspace. The use of our accelerometer sensing gravity can help piece this together.

Figure 2: Circular motion dynamics. Rotating at a constant angular velocity, different length limbs, \( l_1 \) and \( l_2 \), yield different normal accelerations, \( a_1 \) and \( a_2 \).

Figure 3: Gravitation force experienced by an accelerometer at rest. Rotation of the accelerometer can indicate link twist and the plane a link operates within.
This step is probably more helpful to the untrained user. Measurement of the links is not difficult for a human, but the proper architecture of the arm is what requires technical knowledge when implementing inverse kinematics.

Understanding this movement is not only necessary to model the arm, but it is also needed to remove gravity's effect on the accelerometer readings when measuring the length of the links using the method mentioned previously. This is accomplished by subtracting the gravitational vector from the normal acceleration vector at each reading based on the known orientation of the joint and the angle from the encoders.

Putting it Together

We now have the four necessary pieces of data for each link to implement inverse kinematics on the arm (Craig 2005). We are given that the joints have zero offset from one another. We also know how to use the joints, so joint angle data is available. The last two pieces of data were discovered during the automated routines. The length of each link and the link twist of each joint were discovered automatically. Beyond this standard set of data collected for the individual links, we also discovered general information such as how many total links the arm has and where each link is located in comparison to each other. From this data, translation and rotation matrices for use in inverse kinematics can be completed and utilized for the arm.

III. PHYSICAL SYSTEM IMPLEMENTATION

The accelerometer is the central piece to this system. A high degree of accuracy, sensitivity, and speed is required. The VTI SCA3000-D02 3-axis accelerometer was chosen for these characteristics. The use of FC allows it to interface with most any microcontroller or a computer.

A single microcontroller is used to communicate with a PC, control the arm, and access data from the accelerometer. For this we chose a Cypress PSoC 29466 for its flexibility. The microcontroller itself is sending encoder data and accelerometer data to the PC.

Controlling both the arm and the accelerometer from one microcontroller improves the ability to keep data from the two sources synchronized and current. The PSoC also receives control commands from the PC and outputs those to the H-Bridge controlling the proper joint in real time.

The arm we are using is a simple two link, two degree of freedom arm with revolute joints developed in house. It is customizable for in depth testing purposes. The second joint and link can be situated parallel to the axes of the first joint, turned 90°, or removed entirely. The links are thirty centimeters in length and allow plenty of room to adjust the accelerometer for variable length testing. The joints are made of gear head DC motors with high resolution encoders. For initial testing these encoders are crucial for accurate angular velocity data.

The PC receives position data about the arm and returns the control data back to the microcontrollers. It also uses this data to calculate the velocity of the joints differentiating over specified time periods. Then, using data it also received about the normal acceleration, it tracks the length of the arm.

When implementing equation (1) to find the length, care is taken to compensate for slight error in sensors and at times when the arm is at rest. These exceptions can result in divide by zero errors and faulty readings. For this we simply implement a minimum threshold that the angular velocity must meet before calculation can begin.

IV. SINGLE LINK RESULTS

The physical system described has been used for testing, initially using a single link as a proof of concept for link measurements using equation (1) with the given knowns described earlier. This step was also utilized to test the limits of this algorithm, such as the minimum angle needed to calculate the length and the minimum length the system can calculate.

Figure 5 shows a movement to a 90° position. The graph shows angular velocity, normal acceleration, and the length being measured. The lengths the system measures are done in real time and the graph is of these discrete points, calculating the length of the link separately at each time interval. The length calculation (l1 Length) looks mostly smooth and squared off, like a square wave, showing that regardless of the speed or distance, the same...
length is measured instantly when the angular velocity threshold is met. For a final result, this can then be further improved by removing outliers and averaging these points. This measured length compared to actual length gives us very accurate results on our test setup, typically within ±1 cm on a full 30 cm link.

V. TWO LINK RESULTS

These excellent initial results become more difficult when extending it to a multi DOF arm. When an arm is situated in such a way that the first and second joint’s axes of rotation are parallel to each other, disturbance becomes a problem. As mentioned previously, with a single accelerometer setup, joints must remain static while others are moving to calibrate their link length. When a joint is trying to maintain its position and another is trying to move, this is a recipe for disturbance. Figure 6 shows the disturbance received by a joint when another is moving. We see that the joint is pushed approximately 450 encoder clicks (100,000 in a full revolution) and never makes a full recovery. Along the X-axis we see the control output to the H-Bridge barely reacts to the force. Under normal circumstances this could be tolerated, but with sensitive accelerometer measurements, small disturbances become very evident. This small disturbance causes a decrease in the normal acceleration reading for a brief moment, leading to an error in the final results.

Two improvements were made to the control logic in order to reduce the effect of this disturbance on the accelerometer measurement. The first was to design a lead-lag compensator position controller that improves the systems ability to reject noise from the other joint’s torque. Previously a proportional controller had been implemented which was allowed an undesirable amount of disturbance. The second improvement was to use input shaping on the joint in motion so that less disturbance torque is generated at the stationary joint. We can see in Figure 7 that the effects of the disturbance have been significantly reduced over the previous implementation. The joint in the graph is forced back approximately 45 encoder clicks at its highest point (dashed line). Previously it was pushed approximately ten times that amount without a full recovery. The solid line we see is the output to the H-Bridge counteracting the disturbance; previously we could barely see the controller react.

The improved control for this arm allows us to measure multiple links of an arm. The measurement of the second link can be seen in Figure 8. This shows the ability to maintain a steady acceleration (Acceleration) and length measurement (\(L_2\) Length) through a majority of the movement. It is noticeable however the measurements are less steady than those taken with a single link in Figure 5. With the improved control, very little of this is a motor control problem. A total of 50 encoder clicks out of 100,000 (Figure 7) accounts for .18° of disturbance on the motor itself. We are now seeing slop in the gear train of the first joint.

![Figure 6: Disturbance induced on a static joint while another is trying to move and measure itself.](image)

![Figure 2: Disturbance rejection used to increase accuracy of limb measurement. Negative values represent the encoder counts from the force of the disturbance. Positive values show the controller counteracting this disturbance.](image)

![Figure 8: Link 2 Measurement. Notice that the disturbance occurring on joint 1 is still noticeable throughout, but measurement is still possible.](image)
This new control algorithm unfortunately adds complexity to a system that is meant to be automatic with a simple setup. To maintain the automatic setup and intelligent nature of the system the compensator control setup is automated. The computer can simply send an open loop voltage to the motors and utilizing the encoders as one of the knowns, the response will identify the characteristics necessary for proper compensator control. This allows us to continue the trend of automatic setup but with improved results.

VI. FUTURE WORK

The results from the improved two-link disturbance rejection have proven successful in decreasing the disturbance felt on the accelerometer. The controller removes a majority of disturbances received on the encoder. We have now reached the limit of our arm, the remaining disturbances are from gear slop. The next step would be to test this platform on a high-end robot arm to show the true results with multiple links. Additional improvements based on feed forward techniques could be implemented to further reject disturbances and improve accuracy if needed.

The use of this system has proven to work on a simple arm setup, but more robust algorithms could make it useable on a wider variety of arms. The use of the accelerometer could be further expanded to allow for more unknowns.

Assuming that we know where the extended location (zero) for each joint is located could be removed as an assumption. This could be accomplished by implementing new, intelligent algorithms that utilize the accelerometer to look for the maximum acceleration while adjusting a joint, looking for the maximum extended location of the joint. This system currently neglects joints offsets, solving for this piece of the kinematic puzzle would also make the system more intelligent and capable of handling a wider variety of arms.

The use of a single accelerometer was chosen for its simplicity and in an attempt for an elegant and intelligent solution to the problem. We tried to avoid “throwing hardware at the problem”. Some issues with disturbance could be avoided if each joint was also outfitted with its own accelerometer, but provides a less intelligent system.

If a system such as this is perfected there are many possibilities for use in the research, industrial, and educational markets. A humanoid robot could potentially understand how to operate its links without input from a human user. After an understanding of its links, learning how to walk as well as learning new gaits has interesting possibilities. This could also be of use for space exploration robots that are not humanly accessible. A rover may somehow damage a link, maybe from an accident. A predefined kinematic model would then be useless. A dynamically configured kinematic model would have no problem if a link became bent on a distant planet.

VII. Conclusion

The automatic setup and measurement of robot arms is possible using accelerometers. We found that single link results proved very successful and the found two-link measurement possible. Through improvements to the control used on the robot arm, we have yielded better results than previously attainable. We have also been able to detect the orientation of the link and joint with respect to the earth, an important aspect of kinematic fundamentals. The capabilities of an accelerometer provide unique feedback and possibilities not possible with other sensors. This encourages the next step in testing and implementation for further autonomy with this system. Continued improvement may reveal a new platform to some users and new research options available to the experts.

References


