Mechanical Design of a Cartesian Manipulator for Warehouse Pick and Place

M. McTaggart\textsuperscript{1,2}, R. Smith\textsuperscript{1,2}, J. Erskine\textsuperscript{1,2}, R. Grinover\textsuperscript{1,2}, A. Gurman\textsuperscript{1,2}, T. Hunn\textsuperscript{1,2}, N. Kelly-Boxall\textsuperscript{1,2}, D. Lee\textsuperscript{1,2}, A. Milan\textsuperscript{1,3}, D. Morrison\textsuperscript{1,2}, T. Pham\textsuperscript{1,3}, G. Rallos\textsuperscript{1,2}, A. Razjigaev\textsuperscript{1,2}, T. Rowntree\textsuperscript{1,3}, K. Vijay\textsuperscript{1,3}, S. Wade-McCue\textsuperscript{1,2}, A.W. Tow\textsuperscript{1,2}, Z. Zhuang\textsuperscript{1,4}, J. Leitner\textsuperscript{1,2}, I. Reid\textsuperscript{1,3}, P. Corke\textsuperscript{1,2}, and C. Lehnert\textsuperscript{2}

Abstract—Robotic manipulation and grasping in cluttered and unstructured environments is a current challenge for robotics. Enabling robots to operate in these challenging environments has direct applications to industries such as e-commerce, logistics and even agriculture. One of the main challenges associated with these difficult robotic manipulation tasks is the motion planning and control problem for multi-DoF (Degree of Freedom) manipulators. This paper presents the design and performance evaluation of a novel low cost Cartesian manipulator, Cartman who took first place in the Amazon Robotics Challenge 2017. It can perform pick and place tasks of household items in a cluttered environment. The robot is capable of linear speeds of 1 m/s and angular speeds of 1.5 rad/s, capable of sub-millimetre static accuracy and safe payload capacity of 2kg. Cartman can be produced for under 10 000 AUD. The complete design will be open sourced and will be available at the time of ICRA 2018.

I. INTRODUCTION

Enabling robots to pick and place items within cluttered and challenging environments has direct application to industries such as e-commerce, logistics and even agriculture. One of the main challenges associated with these difficult robotic manipulation tasks is the motion planning and control for multi-DoF (Degree of Freedom) manipulators [1]. This can be difficult in scenarios where the environment is cluttered, dynamic and unstructured [2] requiring large amounts of computational time to find a collision-free path in the configuration space of the manipulator. In this paper we argue that designing a manipulator which reduces the complexities of solving the motion planning problem can lead to robust and reliable solutions for real-world deployment.

Robotics competitions are a great driver for developing robotic solutions that are reliable in real world scenarios. Amazon hold an annual competition in which they invite 16 teams from around the world to help solve their warehouse automation problem. The competition requires teams to design a robotic picking solution that can autonomously pick and place a variety of household items. This manipulator design forms part of our winning entry to the Amazon Robotics Challenge 2017 [3].

This research was supported by the Australian Research Council Centre of Excellence for Robotic Vision (ACRV) (project number CE140100016). The participation at the ARC was supported by Amazon Robotics LLC. Contact: matthew.mctaggart@roboticvision.org

1Authors are with the Australian Centre for Robotic Vision (ACRV).
2Authors are with the Queensland University of Technology (QUT).
3Authors are with the University of Adelaide.
4ZZ is with the Australian National University (ANU).

The Amazon Robotics Challenge involves two tasks, stowing and picking. Stowing is the task of picking items from a cluttered tote and placing them into a storage unit [4], emulating the task of storing items received from external suppliers. Picking is the task of taking items from the storage unit and placing them into boxes that are to be delivered to your door. Depending on the storage system, the task essentially requires large vertical motion and horizontal planar motion.

Based on previous experiences, the most common type of manipulators used for this challenge are fully articulated, a UR-5 and Baxter [5] [6] [7]. The main advantage of using an articulated manipulator is the ratio of mechanical footprint to workspace. However, using a fully articulated manipulator comes with the challenges such as singularities [1] in the inverse kinematics which can cause the motion plan to fail or result in unwanted large joint velocities [8]. A bigger issue for robotic pick and place tasks is that the tip position is...
defined by the item to be grasped which give fewer options for the configuration of the arm to achieve a successful grasp. Collisions are likely in this situation. The advantage with Cartesian robots is that they have the same shape irrespective of tool position.

Cartesian manipulators are comprised of three orthogonal linear actuators. If there is no requirement for a small mechanical footprint or the structure of the environment is not complex, such as a mobile manipulation task, then Cartesian manipulators are well suited. Planning times are faster due to simpler inverse kinematics because the configuration space is the task space.

In this paper we propose a cost-effective design of a Cartesian robot (seen in Fig. 1) for pick and place tasks along with extensive performance measures. In particular, the contributions of this paper include:

- The cost-effective design of a 6 DoF Cartesian manipulator.
- Novel differential belt drive system to assist with single motors driving all distal motors.
- Extensive performance analysis verifying the design for use in pick and place tasks.

It features six degrees of freedom; three prismatic joints which form the x, y and z-axes and three revolute joints which form the wrist allowing the end effector to have a roll, pitch and yaw motion. The complete design will be open sourced and will be made available online here: url. This paper presents a brief comparison between Cartesian and articulated manipulators (section II), design (section III) and performance evaluation (section IV) and a discussion of our low-cost Cartesian manipulator. The robot arm has the ability to be used in other scenarios in which a large planar workspace is available, such as fruit harvesting in a greenhouse [9] or factory pick and place tasks.

II. CARTESIAN VS ARTICULATED

Articulated manipulators are common for many different applications due to their versatility and large workspace with respect to their mechanical footprint. The drawbacks are that they can have singularities and discontinuity in trajectory for certain end effector configurations [10] [8]. There are ways of reducing the effect of singularities on motion planning but not eliminating them as shown in the works of [11] and [12].

Using a Cartesian manipulator with a wrist joint to work in a cartesian work space eliminates almost all singularities and reduces discontinuities. A comparison between Cartman and other articulated robots can be seen in Fig. 2. These discontinuities only affect the end-effectors if they try to rotate through them. Since Cartman works in a primarily vertical direction without rotating the wrist joint, these discontinuities do not affect regular operation of the robot. A disadvantage with a cartesian manipulator is the requirement for a larger mechanical footprint in ratio to the overall workspace of the manipulator. This is due to the fact that the linear motion requires some form of support, usually a rail, along the entire length of the axis.

III. DESIGN

III-A Overview

The entire manipulator system is mounted on an aluminium stand shown in Fig. 4. It has three linear axes and a wrist joint which consists of a roll, pitch and yaw axis. Also featured in Fig. 4 is the gripper [13].

III-B Specifications

When designing the manipulator for the Amazon Robotics Challenge task we set out the following specifications:

- A reachable workspace of 1.2m × 1.2m × 1.0m
- A top linear velocity of 1m/s under load along the three linear axes (x,y and z).
- A top angular velocity of 1rad/s under load in the angular axes (roll, pitch and yaw).
- Have a load capacity of 2kg.
- Six DoF at the end-effector, given by three linear axes forming the Cartesian gantry and a three axis wrist.
- Be able to be easily deconstructed/reconstructed for transportation overseas to the Amazon Robotics Challenge event location.

III-C Mechanical Design

Frame: The frame design is constructed out of laser cut and folded 1.2mm sheet aluminium. This was used to keep weight down for transport. The sheet aluminium formed the main outer frame for the system and housed the x-axis belt system and transmission rod. The manipulator frame is mounted on an aluminium stand as seen in Fig. 4.

Motors: Linear motion is performed using Technic’s ClearPath SD-SK-2311S motors. They are a closed-loop brushless motor system designed to be a drop in replacement for stepper motors. By using a stepper-like motor, eliminates the need for external encoders, and makes controlling the linear axes quite simple and similar to how a 3D printer is controlled. They were chosen due their high performance and ease of use.

For the roll, pitch and yaw axes, three Dynamixel Pro L54-50-500 are used. These motors provide the necessary operating torque to hold a 2kg on the end effector whilst under acceleration.

Power Transmission: To actuate the prismatic joints, a belt and pulley system was used. A single motor drives the x axes. In order to eliminate a cantilever effect on the x axes, a transmission rod is used to transmit power from one member to another. One common design that has been observed in a lot of simple manipulator designs, is each axis motor needs to carry the weight of all distal motors as well as the payload. As a result, more powerful motors are needed which then increases weight as well as cost. To solve this problem a differential belt system was designed. Rather than using a single motor to drive a single axes, two
motors work in tandem to drive two axes. The system can be seen in Fig. 3.

**Rails and Bearings:** The linear rails used for the x and y axes are the TBR20 and TBR15 respectively. These are a precision type profile rail which offer a higher precision compared to other linear type rails. Smaller rails were used in the y axis to reduce weight on the system. The y axis consists of two 10mm round rails. Although 10mm rails are small, they offer a light weight solution to the problem. The downside to using 10mm rail however is that when the z axis is extended it creates a pendulum out of the system and induces oscillations at the end effector due to deflection. This was a trade off that was considered during the design process. Although deflection and oscillations are present, steady state accuracy is still achieved with ease once settled.

**III-D Electrical Design**

A single microcontroller is used to control all six axes. The microcontroller that was chosen was the Teensy 3.6 as many libraries exist and has a suitable processor capable of over 180MHz operation. The breakout board needed to include a logic shifter circuits for each of the ClearPath motor pins as the input and output threshold for this microcontroller is only 3.3V and the ClearPath motors needed minimum 5V pulse to operate. In order to interface with the Dynamixel Pro motors, an RS485 module was used so that the Teensy could communicate using the Dynamixel Pro protocol.

**III-E Software Design**

The software can be broken down into two sections: software used on the PC and the firmware used on board the manipulator. A system block diagram of different software interaction can be seen in Fig. 5. Robot Operating System (ROS) [14] was used to handle the high level functionality of the system. Desired end effector poses and robot states were published using the ROS MoveIt! package [15]. This was published as a ROS JointState message type which was processed by the Teensy.

The low level firmware functions send commands to both the ClearPath and Dynamixel Pro motors and also read any feedback that was available from the motors. As the ClearPath motors are a drop in replacement for regular
stepper motors, a stepper motor library was able to be used. The library which was used is AccelStepper [16], a library for use with Arduino compatible devices. This expands on the standard Stepper library provided by Arduino. The main benefit in using AccelStepper is that it allows the user to control the acceleration profile of the motor which is desirable for this application. The Dynamixel Pro were controlled using a modified version of their OpenCR library [17]. The public open source library was developed to be used with their own electronic hardware. Slight modifications were made in order to adapt the library for use with the Teensy 3.6.

IV. PERFORMANCE EVALUATION

IV-A Methodology

In order to test whether the design specs have been met, the motion of the system was tested. Motive, a motion capture system was used to track trajectories during different path plans. The setup can be seen in Fig. 6. Each axis was tested individually at various heights and speeds to test the end point accuracy during motion. Each axis was commanded at a fast and slow speed moving to and from the extremes of the tracking system. The pose of the multi-reflective body seen in Fig. 6 is tracked and interpolated to compare the desired joint position with recorded position whilst in motion and for steady state error.

IV-B Results

A summary of the tests conducted can be found in Table I. Below are the plots of that data that was captured and summarised in these tables.

X-axis: The results of X-axis movements can be seen in Fig. 7. The steady can be seen to converge to close to zero once motion has ceased. The average error in position is 9.62 mm as seen in I. During slow movements, oscillations occur about the desired point due to the pendulum effect but still converge to the desired end point. This is considered sufficient for the application that the manipulator is designed for.

Y-axis: As seen in Fig. 9, the y-axis trajectory oscillates substantially at low heights at slow speeds. This is caused by the pendulum effect of the system as described in section...
Fig. 8. Wrist trajectory at fast and slow speeds. (a) roll-fast, (b) roll-slow, (c) pitch-fast, (d) pitch-slow, (e) yaw-fast, (f) yaw-slow.

### TABLE I

<table>
<thead>
<tr>
<th></th>
<th>Standard Deviation</th>
<th>Mean Error</th>
<th>Static Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>16.07</td>
<td>9.62</td>
<td>6.2e⁻³¹ mm</td>
</tr>
<tr>
<td>y</td>
<td>13.73</td>
<td>9.42</td>
<td>1.9e⁻³¹ mm</td>
</tr>
<tr>
<td>z</td>
<td>13.70</td>
<td>7.91</td>
<td>1.9e⁻³¹ mm</td>
</tr>
<tr>
<td>xzy</td>
<td>64.88</td>
<td>29.63</td>
<td>1.5e⁻³¹ mm</td>
</tr>
<tr>
<td>roll</td>
<td>0.02</td>
<td>0.01</td>
<td>5.6e⁻⁰⁴ rad</td>
</tr>
<tr>
<td>pitch</td>
<td>0.02</td>
<td>0.01</td>
<td>1.5e⁻⁰⁴ rad</td>
</tr>
<tr>
<td>yaw</td>
<td>0.02</td>
<td>0.01</td>
<td>7.3e⁻⁰⁵ rad</td>
</tr>
</tbody>
</table>

Each axis is driven individually and simultaneous motion of axes are also evaluated (xyz). Mean error is defined as the error between the desired position and the measured position. Static error is the error between desired position and measured position once motion has ceased.

IEEE. For the task of the Amazon Robotics Challenge, final point accuracy is more crucial than in-motion accuracy. We can see that the final desired position converges to almost zero. Given this specific task, slow y-axis movements are not necessary any way. The coupling effect of the differential belt system is also of interest. The trajectory of the z-axis while moving the y-axis can be seen in Fig. 10. We can see that the z-axis does travel during pure y-axis motion as a result of the differential belt system. This is to be expected as the two ClearPath motors need to be perfectly in sync in order to reduce this movement to zero. Despite this travel, the final point accuracy is still sufficient for the manipulator’s intended task.

### XYZ axes:
Simultaneous motion of multiple axes were tested against a more complex trajectory. A lemniscate curve was generated on firstly a single plane at three different heights and also in a multi-plane test. The system was tested at different speeds and exhibited the same characteristics as the single axes. Steady state error is almost zero seen in Fig. 12.

**Z-axis:** The z-axis can follow a pure Z trajectory with a fair amount of accuracy as seen in Fig. 11. The z-axis exhibits the least amount of error while in motion with only 7.9 mm mean error in position. As discussed previously, due to the coupling of the Y and Z axes, the y-axis wanders during pure z-axis motion just as the z-axis wanders during pure y-axis motion. Despite this, steady state precision is still achieved.

**Wrist - roll, pitch and yaw:** The wrist joint of the robot exhibit exceptional accuracy not only in motion but in steady state as well as seen in Fig. 8. Achieving a steady state of almost zero error.

**XYZ axes:** Simultaneous motion of multiple axes were tested against a more complex trajectory. A lemniscate curve was generated on firstly a single plane at three different heights and also in a multi-plane test. The system was tested at different speeds and exhibited the same characteristics as the single axes. Steady state error is almost zero seen in Fig. 12.
this design can be used. The maximum target speed was fast movements possibly limiting the applications of which avoid this behaviour, this limits the operating speeds forcing low speeds at approximately speeds of 0.1m/s or less. To effect of the system. We've shown that this only happens at the system in terms of trajectory motion. Y axis motion state error is negligible and can achieve sub-mm and sub-mm accuracy. 

IV-C Verification

Using the data presented in this section, we can verify our speed specifications. Unfortunately due to mechanical limitations, the system experienced belt slip under high accelerations. As a result the acceleration parameter of the controller needed to be lowered. Table II provides a summary of the speed and error achieved by each axis. We've shown that the system is capable of moving a load whilst following linear trajectories with reasonable accuracy. Table I features the static error for each axis. We found that despite oscillations and trajectory error during motion, the steady state error is negligible and can achieve sub-mm and sub-mm accuracy.

V. DISCUSSION AND CONCLUSION

As seen in section IV-B we've shown the limitations of the system in terms of trajectory motion. Y axis motion contributes the most to the overall error due to the pendulum effect of the system. We've shown that this only happens at low speeds at approximately speeds of 0.1m/s or less. To avoid this behaviour, this limits the operating speeds forcing fast movements possibly limiting the applications of which this design can be used. The maximum target speed was not able to be reliably achieved due to belt slippage. As a result, the acceleration of the system was limited yielding a maximum achieved speed of only 0.57 m/s. In future we wish to address this issue as well as the slow movement oscillations by increasing the z-axis rail and bearing system.

REFERENCES

[5] “UR5 - The flexible and collaborative robotic arm.”