

Recycling Robotics: Optimal Motion Planning and Object Ordering

Nicholas Link Samuel Tormey Ricky Levan

December 13, 2014

Abstract

Begin with a cover page that includes your names and an "executive summary" that concisely describes the problem, motivation, your intended solution strategy, and any concerns you have about realizing the objectives. This summary must be 200 words or less and it should be carefully crafted. This is the most important single component of a report, as it is the only one that busy people (like your supervisors) may read. Do not hide critical details or concerns from this summary.

0.1 Problem Statement

What are your project objectives? What need does your project address? Who cares about your results? Which objectives take highest priority? Please also identify the ?applications expert? that is advising you.

Our team is aiming to improve the efficiency of sorting in single stream recycling by creating a robotic movement algorithm to pick items off of a moving conveyor belt. We were inspired to work with recycling because of its positive effect on the environment, and after visiting the a Waste Management Materials Recovery Center (MRF) we found an area that we could address using robotics. Here, we will design an algorithm for a robotic arm to take the maximum number of moving objects off of a moving conveyor belt, which includes solving the optimal path of motion for the arm and the optimal ordering of objects to be grabbed.

The MRF sorts materials in single stream recycling, and in one section employees sort clear and colored plastics. Even though sorting colors of plastics generates more revenue for the MRF, often these plastics are not sorted because of the necessary manual labor. We believe that parts of the sorting process such as this one can and should be automated for the sake of the reduced labor costs and economic gains for the MRF . Making recycling more economically efficient is important because recycling saves energy and is therefore important for the environment.

In the MRF, the plastics move on conveyor belt with the other recycling materials, and there already exists devices that can detect the location of colored and non-colored plastics. Therefore, our robotic arm will take as input the objects' locations on the belt and then grab the maximum number of objects and place them above or off the belt. This consists of solving the optimal path for the robotic arm to take to reach an object as well as solving the optimal ordering of objects to be picked so as to maximize objects grabbed and minimize time used. We will use a 3-degree of freedom arm, with 2 revolute joints in the plane and one prismatic joint for the z-axis. This will allow the robot to reach anywhere in it's reachable workspace on the belt and reach down in between objects in order to grab the plastics that need to be taken off the belt. If we create a successful algorithm for a single robotic arm then we will look into algorithms for multiple robotic arms. We are hoping that our robotic arm algorithm could be implemented in the Houston MRF and other MRFs to improve the efficiency of recycling.

Applications expert?

0.2 Literature Review

How have others approached this problem? Include bibliographic citations. This should be a bit more concise than the literature reviews in Tech Memo 2; you should also add sources that you have discovered since writing that memo. Please use citations and include them in a bibliography at the end of the document.

0.3 Design Criteria

Clearly explain your design criteria, and how each is quantified. In addition to the material from Tech Memo 3, include here the criteria weighting used in Tech Memo 5. We split our project into three subproblems and then identified and quantified criteria for each subproblem. The three subproblems are the decision-making algorithm, minimal time motion planning, and the real demonstration. Finally, we weigh the criteria for each subproblem so that the sum of the weights adds up to one.

Below we list the criteria for each subproblem. The criteria name is in bold, the weight is in parenthesis on the left, and the description is to the right of the criteria name.

0.3.1 Decision-Making Algorithm

1. **(25%) True positive grabs:** (Number of objects we intend to grab that we actually grab, divided by the number of objects that we intend to grab) We aim for a 5% improvement rate on a first-in first-out (FIFO) algorithm
2. **(20%) Impurity of collection:** (Number of objects we did not intend to knock off that we knock off, divided by the number of objects that we knock off) We aim for this to be less than 5%
3. **(20%) Novelty:** Come up with 1 new algorithm
4. **(15%) Online Computational Time:** Able to make decision for next object in less than 1/10th of a second
5. **(15%) Software Bugs:** 0 major stalls
6. **(5%) Cost:** Spend less than \$100 on software

0.3.2 Minimal Time Motion Planning

1. **(40%) Control Time:** Average of 10% faster than some to-be-determined naive motion planning
2. **(30%) Novelty:** One new minimal control calculation for our specific robot
3. **(15%) Software bugs and stability:** 0 major stalls or crashes
4. **(10%) Online Computational Time:** Online calculation takes less than a tenth of a second
5. **(5%) Cost of software:** Less than \$100

0.3.3 Real Demonstration

1. **(25%) Time Accuracy:** (Real motion time / Theoretical optimal motion time): 95%
2. **(25%) Positional accuracy:** Average error within 2 centimeters
3. **(20%) Hardware malfunction:** Malfunctions such as hitting the ground, hitting itself, gripper malfunction, electronics malfunction. Spend less than \$50 and 20 hours fixing with hardware
4. **(10%) Accurate representation of MRF:** 1 (unrealistic) to 5 (MRF), rated at least a 3 by at least 3 professionals and academics in relevant companies or fields
5. **(5%) Safety:** 0 Injuries from swinging robotic arm
6. **(5%) Material Cost:** Robot costs less than \$300, camera costs less than \$50, conveyor belt costs less than \$100, and we will spend \$0 on storage
7. **(5%) Size:** Entire system is contained in a space of 1-3 cubic meters
8. **(5%) Captivating:** On a 1 to 5 scale, people visiting the demo rate us on average at least a 3

0.4 Candidate Design Solutions

Highlight the main strategies that you described in Tech Memo 4. If you included substantial mathematical content in that memo, scale that back here, especially for designs that you have since ruled out as not viable. (It is appropriate to give mathematical content for the main solutions that you intend to implement, including a description of numerical methods and software required.)

0.5 Design Evaluations

Include your scoring matrices from Tech Memo 5, along with your supporting rationale. Clearly specify the approach that you intend to implement. Specify the mathematical experts you have identified to advise you in implementing this solution.

Below we rank the proposed solutions to our subproblems according to the weighted criteria. We ranked the solutions according to each criteria from 1 to 5 (5 being the highest). Then we scaled the ranking to the criteria weights and took the sum of the scaled scores for each solution. The solutions that won are an analytic motion planning solution, the SPT zoned and entry-biased decision making algorithms, and the InterbotiX Labs WidowX Robot Arm for the real demo.

0.5.1 Motion Planning

		Analytic		Own Numeric Solver		Existing Numeric Solver	
control speed up	.40	5	2.00	3	1.20	4	1.60
novelty	.30	4	1.20	5	1.50	1	0.30
bugs	.15	2	0.35	2	0.30	5	0.75
comptime	.10	5	0.50	1	0.10	3	0.30
cost	.05	5	0.25	5	0.25	1	0.05
<i>score</i>		4.25		3.35		3.00	

The analytic solution will provide the exact answer for the shortest path while the numeric solvers will only approximate. Creating our own numeric solver will be the most novel, while finding an analytic solution will still be novel. The analytic solution will be challenging and we are likely to make mistakes (equivalent to bugs), and our solver is much more likely to have bugs than an existing numeric solver that has been through rigorous testing.

The analytic solution will probably require very little computational time, while our solver may take a lot of time due to inefficiencies. Finally, an analytic solution or making our own solver will likely cost nothing while using an existing solver may cost money.

0.5.2 Decision Making Algorithm

		SPT Zoned		Combinatorial Optimization		Entry Biased SPT		Genetic Algorithm		Adaptive Algorithm	
True Positives	.25	3	0.75	5	1.25	3	0.75	3	0.75	4	1.00
False Negatives	.20	3	0.60	3	0.60	3	0.60	3	0.60	3	0.60
Novelty	.20	4	0.80	2	0.40	4	0.80	3	0.60	4	0.80
Bugs	.15	4	0.60	2	0.30	4	0.60	2	0.30	3	0.45
Comp. Time	.15	5	0.60	1	0.15	5	0.75	2	0.30	2	0.30
Cost	.05	3	0.15	3	0.15	3	0.15	3	0.15	3	0.15
<i>score</i>		3.65		2.85		3.65		2.70		3.30	

The combinatorial optimization method will likely compute the exact best order while the other algorithms provide only good estimates, and adaptive is ranked in between the two because it follows combinatorial optimization part of the time. All of the algorithms are probably equally likely to pick up false negatives. The novelty of the algorithms is based on how similar they are to algorithms we found in literature. We believe the simpler algorithms are less susceptible to bugs, while more complex algorithms like combinatorial optimization and the genetic algorithm are more susceptible. As well, the computational time of the simpler algorithms is likely to be much shorter. Finally, we are not sure which of these algorithms would require buying software so we ranked them equally with respect to cost.

0.5.3 Real Demo

		Adept Cobra i600 Scara		InterbotiX Labs WidowX Robot Arm		SainSmart 4-Axis Robot Arm Model	
Position Accuracy	.25	5	1.25	5	1.25	4	1.00
Time accuracy	.25	5	1.25	5	1.25	3	0.75
Hardware Malfunction	.20	1	0.20	5	1.00	3	0.60
Acc. Rep. of Rec. Plant	.10	5	0.50	1	0.10	3	0.30
Cost	.05	1	0.05	1	0.05	5	0.25
Safety	.05	2	0.10	5	0.25	5	0.25
Captivating	.05	5	0.25	5	0.25	3	0.15
Size	.05	1	0.05	4	0.20	5	0.25
<i>score</i>		3.65		4.35		3.55	

The Adept Cobra i600 Scara (AC) and InterbotiX Labs WidowX Robot Arm (IL) are higher quality robots than the 4-Axis Robot Arm Model SainSmart (SS) so they are more likely to be accurate with respect to time and position. The AC is old and very difficult to work with so it is likely to have hardware malfunctions while the IL and SS are less likely, although the SS is ranked in the middle because of its quality. The AC most accurately represents what could be implemented at the MRF, the IL does not represent the MRF, and the SS is an OK representation of the MRF. The AC and IL are expensive, while the SS is cheap, so it ranks highest with respect to cost. Because of the difficulties of working with the AC, we ranked its safety low while the safety of the IL and SS is ranked higher. The AC and IL are the most captivating because of their quality while the SS is less interesting. Finally, the AC is large while the IL and SS are relatively small.

0.6 Preliminary Prototyping/Results

All teams should now have begun prototyping solutions. Please describe your progress. If you have preliminary results from basic models, please include those results.

0.7 Project Plan for Spring Semester

This is a vital new component of the report. You should provide a detailed time-line that breaks your project into smaller subtasks, showing the period of time over which you intend to tackle each subtask. Please illustrate this graphically using a Gantt chart broken into weekly time intervals covering the Spring semester. (It

would also be helpful if you sketch out how you plan to break up the work among your group members.) Your final project presentations will be made mid April 2014 in the CAAM Lunch seminar; your final project reports are due end of April 2014. (Exact dates will be determined in Spring 2014). You can draw Gantt charts using a variety of online tools (e.g., tomsplanner.com).