

# Lessons from the Amazon Picking Challenge: Four Aspects of Building Robotic Systems

Clemens Eppner\* Sebastian Höfer\* Rico Jonschkowski\* Roberto Martín-Martín\*  
Arne Sieverling\* Vincent Wall\* Oliver Brock

**Abstract**—We describe the winning entry to the Amazon Picking Challenge. From the experience of building this system and competing in the Amazon Picking Challenge, we derive several conclusions: 1) We suggest to characterize robotic system building along four key aspects, each of them spanning a spectrum of solutions—modularity vs. integration, generality vs. assumptions, computation vs. embodiment, and planning vs. feedback. 2) To understand which region of each spectrum most adequately addresses which robotic problem, we must explore the full spectrum of possible approaches. To achieve this, our community should agree on key aspects that characterize the solution space of robotic systems. 3) For manipulation problems in unstructured environments, certain regions of each spectrum match the problem most adequately, and should be exploited further. This is supported by the fact that our solution deviated from the majority of the other challenge entries along each of the spectra.

## I. INTRODUCTION

The *Amazon Picking Challenge* tested the ability of robotic systems to fulfill a fictitious order by autonomously picking the ordered items from a warehouse shelf (Fig. 1). The system presented here outperformed the 25 other entries, winning by a significant margin. In this paper, we provide a detailed technical description and experimental evaluation of our system. We also present three main conclusions from our system-building experience:

- 1) Robotic systems can be characterized along *four key aspects*. Each of these aspects can be instantiated by selecting from a *spectrum of approaches*.
- 2) To develop a shared understanding of system building, i.e. which region of each spectrum most adequately addresses a particular robotic problem, we should *explore these spectra* and characterize systems based on them.
- 3) For *manipulation in unstructured environments*, we believe that certain regions of each spectrum match the problem characteristics most adequately and should therefore be examined by roboticists with increased emphasis.

We gratefully acknowledge the funding provided by the Alexander von Humboldt foundation and the Federal Ministry of Education and Research (BMBF), the European Commission (SOMA project, H2020-ICT-645599), the German Research Foundation (DFG) (Exploration Challenge, BR 2248/3-1), and the travel grant provided by Amazon Robotics. We would like to thank Barrett Technology for their support and our team members Raphael Deimel, Roman Kolbert, Gabriel Le Roux, and Wolf Schaarschmidt, who helped creating the winning system.

All authors are with the Robotics and Biology Laboratory, Technische Universität Berlin, Germany.

\* Authors contributed equally to this work and are ordered alphabetically.



Fig. 1. Our robot picks a “plush duck toy” during the challenge.

The four key aspects of system building are:

**A. Modularity vs. Integration:** In robotics, the behavior of the entire system determines success, not the performance of individual modules [8]. Still, a high degree of modularity allows breaking down problems into simpler subproblems, which is especially useful when the overall problem is too complex to solve. Wrong modularization, however, can make solving problems unnecessarily difficult. Until we fully understand which modularization is most adequate for manipulation in unstructured environments, we suggest to build tightly integrated systems and constantly revise their modularization [27].

**B. Computation vs. Embodiment:** Robot behavior results from the interplay of computation (software) and embodiment (hardware). Computation is a powerful and versatile tool but adapting the embodiment sometimes leads to simple and robust solutions. We suggest that in manipulation, one should consider alternative embodiments as part of the solution process, so as to most synergistically match software and hardware.

**C. Planning vs. Feedback:** Planning performs search in a world model, leading to verifiable solutions. Feedback from physical interactions, on the other hand, reduces uncertainty and allows to find local solutions without expensive computation. We thus suggest to use planning only when necessary and explore the use of feedback as an alternative when the manipulation task does not require global search.

**D. Generality vs. Assumptions:** For robotics research, finding general solutions is highly desirable because they apply to a wide range of robotic tasks. However, solving the most general problem might be unnecessary or even unfeasible.

We suggest to search for reasonable and useful assumptions that aid solving manipulation problems in unstructured environments. By extracting, sharing, and revising assumptions that prove useful for an increasingly broader variation of a problem, we will naturally progress towards a general solution.

These aspects are not novel and certainly will not surprise the robotic practitioner. However, what should come as a surprise is the sparsity with which the corresponding spectra have been explored by our community and how rarely these aspects are used explicitly to characterize robotic systems. Case in point: Our solution to the Amazon Picking Challenge explores different regions on these spectra than most other challenge entries [13]. We believe—and will support this belief in the remainder of this paper—that these differences were crucial for our success.

We propose that by making the four key aspects of robotic systems (and possibly additional ones that we did not identify yet) explicit, our community will begin to understand the mapping of problem characteristics to the appropriate regions on these spectra. Our paper is, of course, only a single data point in this endeavor. But if our community starts characterizing robotic systems along the proposed axes, thereby making design choices transparent and comparable, we might move towards a scientific theory of system building [2, 20, 27].

In Section III, we present the specific choices we made for our system and relate them to the four aspects. Section IV then describes the performance of the system both in the competition and in additional experiments conducted in our lab. We then discuss the four aspects in Section V in a first attempt to gain insights into the relationship between problem type and the most appropriate region of the spectra. Finally, in Section VI, we discuss the implications for problems in mobile manipulation, of which we consider the Amazon Picking Challenge to be an instance.

## II. THE AMAZON PICKING CHALLENGE

In the Amazon Picking Challenge, the task consists of autonomously picking twelve out of 25 objects (Fig. 2) from a warehouse shelf and placing them into a storage container (Fig. 1) within 20 minutes. The robot has knowledge of which objects are contained in each of the shelf’s twelve bins, but not of their exact arrangement inside the bin.

a) *Evaluation Criteria:* For each successfully picked target object, the robot receives 10, 15, or 20 points, depending on how many additional objects were in the same bin (between none and three). Objects that were considered difficult to pick obtained up to three bonus points. Picking the wrong object results in -12 points.

*b) Objects:* The 25 competition objects varied widely in size and appearance (Fig. 2). Objects ranged from small spark plugs to bulky boxes, from hard cases to soft plush toys, and from loose objects to those wrapped in plastic. This variety presented a challenge for grasping and perception.

c) *Environment*: The items were placed in the twelve different bins of the shelf. The robot was allowed to operate in a  $2\text{ m} \times 2\text{ m}$  area in front of this shelf. The bins had a rather



Fig. 2. The 25 picking objects from the challenge

small opening ( $21\text{ cm} \times 28\text{ cm}$ ) but extend quite far to the back ( $43\text{ cm}$ ). The small opening was problematic for many state-of-the-art robotic hands as they barely fit into the bins, once the shelf is stuffed. As a result, motion capabilities within the bins were greatly reduced. Since objects could be surrounded by other objects or could lie close to the side wall of the bin, grasping opportunities were restricted. Moreover, the bins had a lid in the front, providing an opportunity for the robot or the grasped object to get caught while retracting from the bin.

The challenge environment also posed significant challenges for perception. Due to the narrow bins, objects were often visible from one side only and partially obstructed. The floors of the shelf were made of reflective metal, rendering color and depth information unreliable. During the challenge, the lighting conditions were particularly difficult due to very bright spot lights directly above the competition area: objects in the front of each bin appeared to be effectively white, while objects in the back appeared as nearly black.

### III. TECHNICAL SYSTEM DESCRIPTION

We now describe the hardware and algorithmic components of our solution. We will mention connections between our design choices and the four key aspects. In Section V, we then provide a more detailed discussion of these choices and the related trade-offs along the four spectra.

### A. Hardware Components

Our robot is composed of a mobile manipulator, a custom suction-cup gripper, and various on-board sensors (Fig. 3).

1) *Mobile Manipulator*: We use a seven degree-of-freedom Barrett WAM [4] mounted on a Nomadic XR4000 mobile base [53]. Four caster wheels in the base with eight actuatable axes provide holonomic motion capabilities [22]. The entire kinematic chain possesses ten holonomic degrees of freedom.

The inclusion of (holonomic) mobility—a choice of embodiment that set our solution apart from most other entries in the Amazon Picking Challenge—greatly facilitated the generation of motion. The ability to reposition the base enabled the arm to easily reach inside all of the bins, leading to simpler arm motion than with a static base. The increased dimensionality of the configuration space poses challenges for motion planning;

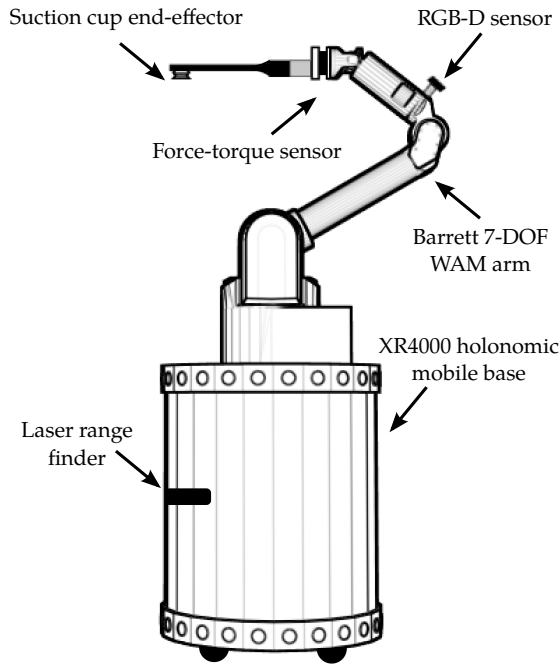


Fig. 3. Schematic drawing of the robot

we avoided these challenges by generating the robot’s motion from pre-defined sequences of joint- and task-space feedback controllers (Sec. III-B). This simple, yet effective solution is an example of how appropriate choices for embodiment (Sec. V-B) and the use of feedback (Sec. V-C) lead to effective solutions.

2) *End-Effector*: Our end-effector consists of a modified crevice nozzle from a vacuum cleaner with a suction cup mounted at its tip (Fig. 4). An off-the-shelf vacuum cleaner, operating at 250 W, generates sufficient air flow (and suction in the case of a tight fit between suction cup and object) to lift up to 1.5 kg.

This simple end-effector can reliably pick all but one of the challenge objects (the pencil cup) from the narrow, deep shelf bins. Grasping success is rather insensitive to the exact contact location with the object, leading to reduced requirements for perception. At the same time, the end-effector’s thin shape reduces the need for complex collision avoidance or pre-grasp object manipulation, as it easily fits in between objects, pushing them aside if necessary. This simple choice for the end-effector illustrates that an appropriate embodiment simplifies different aspects of the overall solution, including grasping, grasp planning, and perception (Sec. V-B).

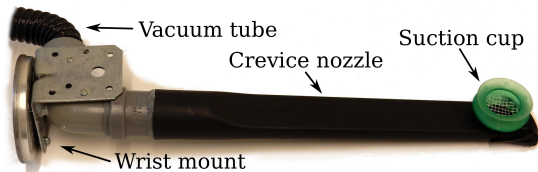


Fig. 4. The suction cup end-effector

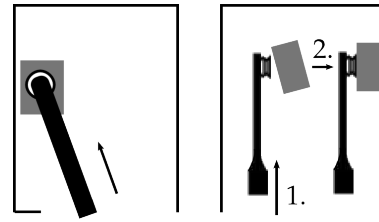


Fig. 5. Top-down view of the our picking primitives: Top-down picking, avoiding the left frontal side lip (left); side picking, aligning the object (grey) with the right wall (right)

3) *Sensors*: All sensors are on-board: a base-mounted SICK LMS-200 laser range finder, an Asus XTion Pro RGB-D camera on the robot’s forearm, a six-axis ATI Gamma force-torque sensor on the wrist, and a pressure sensor inside the tube connecting the vacuum cleaner to the end-effector. These sensors provide feedback to monitor and guide task execution and to reduce uncertainty (Sec. V-C). The robot uses the laser range finder for particle-filter-based localization with respect to a 2-D map of the shelf’s base, it uses the force-torque sensor to guide the end-effector motion, and the pressure sensor to detect picking failure.

## B. Motion Generation

After sorting the list of target items in decreasing order of expected number of points to be gained, the robot chooses between two picking strategies:

- *Top-down*: The end-effector is positioned above the object, moves downward until a force measurement signals contact with the object and activates suction. The protruding metal edges on the left- and rightmost bins of the shelf are avoided by slightly rotating the end-effector around the vertical axis in these bins (Fig. 5, left).
- *Side*: The end-effector approaches the object from the left or right. The robot executes a force-guarded motion orthogonal to the bin wall, pushing the object to the side until it senses contact, and then activates suction (Fig. 5, right).

Both grasping primitives deliberately move the end-effector *into* the object and push it against the floor, walls, or other objects. Aligning the objects this way simplifies the picking action. This is an example of exploiting the environment to guide manipulation [17] using haptic feedback (Sec. V-C).

To select one of the two primitives, we developed a scoring method that estimates their chance of success. The score is based on (i) how well the given side of the object can be grasped and (ii) the amount of free space to bring the end-effector into position. For this, the robot determines the orientation of objects by estimating their bounding box (Sec. III-C). This is important, for example, in the case of books which must be picked up from the front or back cover.

This scoring scheme is an example of how to facilitate the solution, in this case for pick planning, by leveraging prior knowledge about the problem (Sec. V-D).



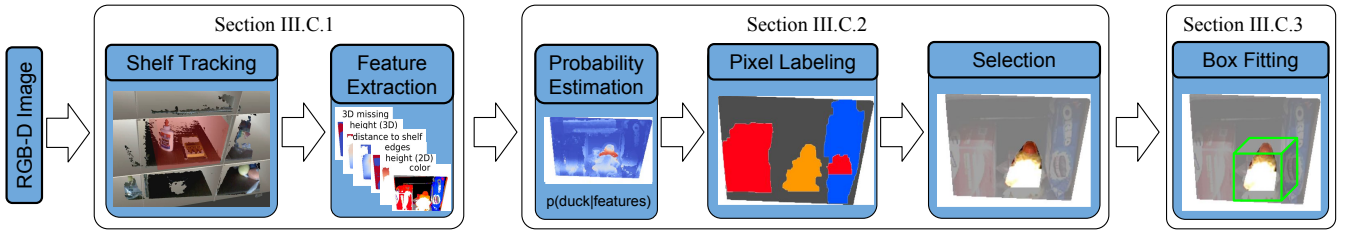


Fig. 7. Object recognition pipeline (from left to right): We estimate the region of the image containing the order bin by tracking the shelf in the depth map; we compute a set of features per pixel of the region; we estimate the probability that a pixel belongs to an object; we assign object labels to each pixel; the target object is segmented in the image; a bounding box of the object size is fitted to the image segment.

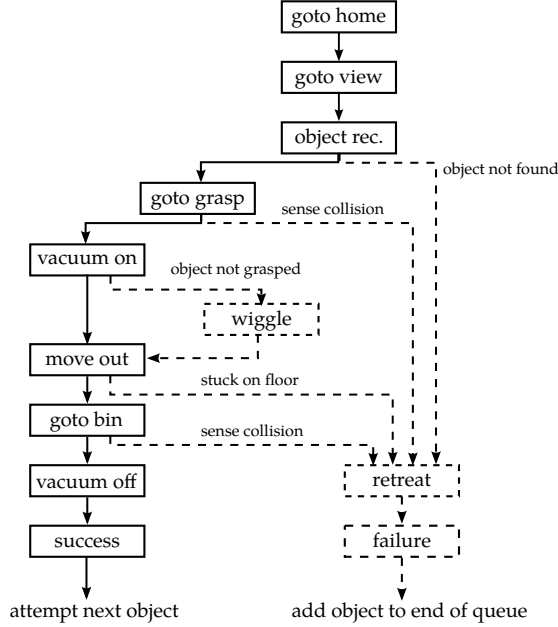


Fig. 6. Simplified hybrid automaton for one picking primitive: The dashed nodes and edges handle failures during execution.

The execution of picking motions is realized with continuous feedback controllers. We transition between these controllers based on discrete sensor events. This behavior can be described by hybrid automata [16]. States in a hybrid automaton correspond to one or several feedback controllers, while state transitions are triggered by sensor events. For each object, we generate and execute a hybrid automaton that implements the selected picking primitive (see example in Fig. 6). The automaton describes the actions for a successful grasp but also contains countermeasures for common failures. The resulting hybrid automaton consists of 26 states and 50 transitions, of which 34 deal with error handling. Error handling occurs most frequently in response to sensor input. For example, if the robot detects an undesired contact with the shelf, it retracts from the bin and reattempts to pick the object later.

The hybrid automaton not only enables us to efficiently exploit feedback (Sec. V-C) but also to compose motions defined in different spaces, such as joint space or task space. This permits combining precise and repeatable joint-space

controllers (for moving the arm-mounted camera to a predefined view position), task-space controllers [37] involving base and arm (for reaching in and out of the shelf), and task-space motion involving only the arm (for fine-positioning inside the shelf).

### C. Object Recognition

To successfully pick specific objects, the robot must recognize and locate the target object. Our system captures images with an RGB-D camera mounted on the robot's forearm and performs three steps: feature extraction, object segmentation, and bounding box fitting. The first step extracts a number of task-specific features for every pixel of the RGB-D image. Statistics about these pixel-features for each object enable the second step to find the image segment that has the highest probability of belonging to the target object. The third step takes the point cloud for this segment and fits a bounding box with size of the target object to this point cloud. This bounding box allows the robot to decide where and from which direction it should perform the pick. Fig. 7 shows how the individual steps connect. We will now examine these three steps in more detail and relate them to the key aspects of robotic systems. We have also published a more detailed description of our object recognition [26].

1) *Feature Extraction*: To extract features from the target bin, the robot tracks the shelf in the depth image using the iterative closest point (ICP) method. Based on the tracking, it crops the RGB-D image to only show the target bin and computes the following six features for each pixel:

- a one dimensional color representation, equal to the hue for saturated pixels and one of three distinct values for white, gray, and black for unsaturated pixels
- a binary visual edge feature
- the distance to the tracked shelf model
- the height above the ground plane of the shelf bin
- the height in the image plane (for pixels without depth information)
- a binary feature about the presence/absence of depth information

These features discriminate between most objects and the shelf. Instead of searching for features that could solve the general object recognition problem, these task-specific features rely on strong assumptions (e.g. that objects are placed in a known shelf) in order to simplify perception. As a result of

these assumptions, our perception system was able to handle the unusual lighting conditions during the Amazon Picking Challenge, non-rigid objects, partial views, and partial depth data. By making suitable assumptions, our approach outperforms generic off-the-shelf methods in the specific settings of the Amazon Picking Challenge [41], [26] (Sec. V-D).

2) *Object Segmentation*: Based on manually segmented training images, our method precomputes histograms for each feature and object. It uses these histograms to estimate how likely each feature value in the current image is for each object. By multiplying these likelihoods and normalizing them per pixel, we compute the probability of each pixel in the current image to belong to a specific object. We compute these probabilities for all objects that we know are contained in the target bin and for the bin itself. Our method then smooths these probability images, labels each pixel corresponding to the most probable object, and selects the segment that includes the pixel with the maximum probability for the target object.

This step exploits additional assumptions to facilitate segmentation, for example that only a small subset of all objects is present in every bin and that the robot knows which objects these are (Sec. V-D).

3) *Bounding Box Fitting*: The segment estimated by the previous step is now transformed into a point cloud representation, filtered for outliers, and used to fit a bounding box. The fitted bounding box is then compared to the true dimensions of the target object to match the sides of the object correctly. The result of this step is an approximate estimate of position and orientation.

Note that bounding box fitting is tailored to the requirements of our picking strategies. The embodiment of the robot (Sec. V-B), in particular the design of the end-effector, does not only simplify the picking motion but also relaxes requirements on exact pose estimation of the target object. Picking is successful as long as the end-effector makes contact with a pickable side of the object. This combination of embodiment and algorithmic choices illustrates the advantage of tight integration (Sec. V-A) in contrast to premature modularization.

#### IV. EVALUATION

The Amazon Picking Challenge provides an in-depth evaluation of our system, comparing its performance with 25 teams from around the world in real-world conditions. We complement the competition results with nine additional experiments, using all object configurations used during the competition. We also discuss capabilities and limitations of our system.

##### A. Quantitative Evaluation

1) *Competition Result*: In our competition performance, we scored 148 out of 190 points (77.8%). We attempted to pick all twelve objects and were successful for ten. An average picking motion took 87 seconds. This allowed us to maximally attempt 14 picks within the 20 minutes of challenge duration. The scored points put us well above the competitors who scored 88 points for second, and 35 points for third place.

2) *Post-Hoc Evaluation*: To gather more data about our system’s performance, we reenacted all five shelf configurations that were used in the challenge. We performed two trials per setup, using the robot system from the challenge without modifications. The total testing time was 200 minutes in which the robot picked 95 objects, of which 85 were target objects. We reached an average point count of 117.6 ( $\sigma = 29.2$ ) which is 62.5% of all available points. This shows that the competition run was on the upper end of the system’s capabilities. Still, out of the ten trials, only one (72 points) would have lead to our team placing second.

##### B. System Capabilities

*Object type*: Our robot can pick 24 of the 25 objects, irrespective of whether they are soft, rigid, heavy, or light. It cannot pick the pencil cup, as its meshed structure foils the suction-based gripper (Fig. 8d, Sec. IV-C5).

*Object placement*: The system is able to execute all grasp types in all bins and can even pick up objects close to the back wall.

*Lighting conditions*: The object recognition is invariant to extreme lighting condition variations, such as the ones encountered during the competition.

*Shelf pose*: The system can handle displacements of the shelf with continuous feedback from localization.

*Shelf collision*: Collisions with the shelf occurred during the tests but they do not lead to system failure.

*Long-term operation*: During the 200 minutes of running time, we did not encounter a single total system failure, although we encountered several unexpected events in the experiments (next section). We attribute this to the significant amount of time we invested in adversarial testing, leading to the many failure cases covered by the hybrid automaton (Fig. 6).

##### C. System Limitations

From the 120 attempted picks, the system picked ten wrong objects and failed to pick 25 objects. Table I shows the six failure categories. We will discuss these failures in detail and find that all failure cases can be addressed by shifting along the spectra of the proposed aspects.

85	successful picks
13	object recognition failures
9	bulky objects stuck at removal
9	missing small objects due to end effector imprecision
2	displacing objects during approach
2	meshed pencil cup (cannot be picked with our end-effector)

TABLE I  
FAILURE CASES FOR 120 PICKING ATTEMPTS

1) *Object Recognition Failures*: Perception was one of the main challenges in the competition. We attribute 13 failed picking attempts to the object recognition pipeline. These failures occur when our local features cannot discriminate between the objects present in the target bin, resulting in wrong object boundaries (Fig. 8a) or mistaking another object for

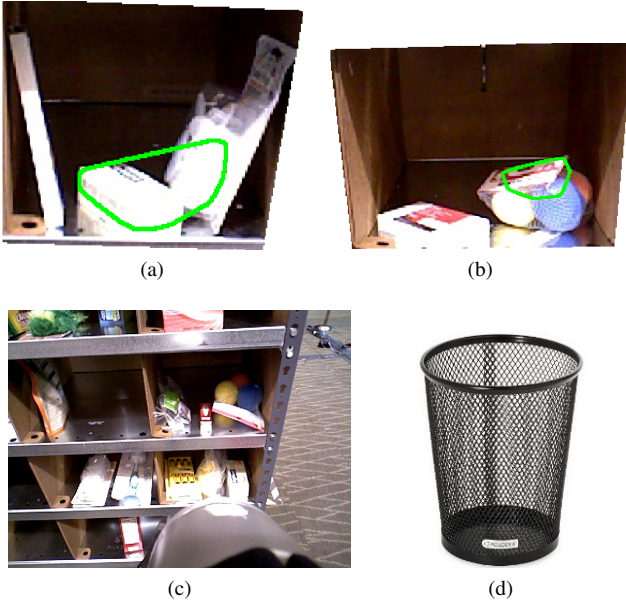


Fig. 8. Failure cases of our system: (a) Wrong object boundary (b) Mistaking part of another object (right) for the target object (left) (c) The plush balls (right) were picked instead of the small spark plug (middle). (d) The pencil cup cannot be picked up with our end-effector.

the target object (Fig. 8b). Consequently, the robot sometimes chooses the wrong pre-pick pose or the wrong picking primitive.

We believe that object recognition can be improved most effectively by shifting along spectrum A towards tighter integration with the other system components (Sec. V-A), and along spectrum C towards more feedback (Sec. V-C). For example, we could reject poses that result in physically implausible configurations by tighter integration of segmentation and geometric pose reasoning. Moreover, we could decrease the likelihood of picking wrong objects by incorporating additional feedback, for example, by weighing objects or visually inspecting them after the pick.

2) *Bulky Objects Stuck at Removal*: Eight scenarios contained a large box which could only be removed from the bin by tilting it. Our system failed on all eight attempts. The long bottle brush also got stuck once on the shelf lip once and dropped in the process of removal. To address these failures, we need to shift along spectrum C towards (motion) planning (Sec. V-C). Planning would allow us to reason how to reorient objects to remove them from the bin.

3) *Small Objects*: Out of ten attempts, the robot failed nine times attempting to pick up the small spark plugs. In the competition run, the robot even picked up a non-target object instead (Fig. 8c). These failures result from the fact that the reaching movement is executed open-loop, accumulating a significant error in forward kinematics of the arm, resulting in a pose error of up to 1 cm. This can be addressed easily by shifting along spectrum C towards more feedback (Sec. V-C), for example by using visual servoing.

4) *Displacing Objects*: In five out of ten attempts, the robot toppled over the glue bottle. The bottle then required

a reattempt from the top. In two cases the robot did not have enough time for a reattempt and lost points. As before, this failure case can be alleviated by additional feedback; the robot could detect the tumble and immediately change the strategy.

5) *Pencil Cup*: The meshed metal pencil cup (Fig. 8d) does not have enough solid surface to pick it with suction. This failure mode shows a limitation of our chosen embodiment (Sec. V-B). It suggests possible extensions to our end-effector, e.g. adding a mechanical or magnetic gripper.

## V. FOUR KEY ASPECTS OF BUILDING ROBOTIC SYSTEMS

We will now generalize our experience from the Amazon Picking Challenge to building robotic systems in general. We structure this discussion along the four key aspects and their respective spectra. For each aspect, we outline the range of approaches by presenting arguments and examples for both ends of the spectrum. Then we position our system on this spectrum.

### A. Modularity vs. Integration

There is a continuum between tightly integrated and modular solutions. This continuum has been previously investigated in the fields of systems engineering, computer science and product management [3, 49].

1) *Modularity*: Building systems of arbitrary complexity without structuring them into modules is very difficult. Modularity is a way of decomposing complexity by breaking down a problem into smaller sub-problems that can be solved and tested individually. Furthermore, modules with defined interfaces allow to use, replace, compare, and recombine existing modules to solve new problems. For these reasons, building modular systems is the prevalent paradigm in robotics. This is reflected in the separation of robotics into the classical fields of perception, planning, control, etc. as well as in the produced software. For example, high modularity is one of the core concepts of ROS [40], a popular framework for implementing robotic systems. Similarly, libraries like OpenCV [7], PCL [43], and MoveIt [47] represent commonly employed modules for computer vision and planning, offering stable interfaces and well-tested functionality.

2) *Integration*: Robotic systems generate behavior as a result of integrating many software and hardware components [8, 11]. Therefore, the usefulness of a robotic system is determined by the performance of the integrated system, rather than by the performance of individual components. To ensure that the performance of the entire system is maximized, and to avoid making wrong commitments or addressing sub-problems that are unnecessarily difficult, all components of the system should be chosen to maximally exploit potential synergies between components. To identify these synergies in the absence of established system-building guidelines requires early integration [24, 27].

Many important advances in robotics research were achieved by overcoming existing “modularizations” and the corresponding boundaries between sub-fields of robotics. For example, SLAM [48] couples localization and mapping in a

recursive estimation loop to solve the joint problem more effectively. More recently, combining interaction and perception led to advances in robot perception [6, 27, 34].

3) *Our Design Choice on the Spectrum:* Our system for the Amazon Picking Challenge used ROS [40] and relied on various standard modules, for example, for visual processing [7, 43] and navigation [32]. However, we embraced tight integration at various levels. For example, we integrated planning and control using hybrid automata, adapted our picking strategies to the embodiment (Sec. III-B), and the requirements for object recognition to the picking strategies (Sec. III-C).

Our development process was tailored to building a tightly integrated system, by adapting many ideas from agile development [45]: rapid prototyping, early and continuous integration, adversarial testing, and shared knowledge.

### B. Computation vs. Embodiment

Aspect B describes the degree to which a robotic system relies on explicit computation (software) or on the embodiment of the robot (hardware). The idea that mechanisms and materials can take over some of the processes normally attributed to computation is known as *morphological computation* [39].

1) *Computation:* Since computation is more flexible and can be altered easily, compared to the embodiment (hardware), focusing on computation allows building highly complex systems with diverse behaviors in a short amount of time. Purely computational approaches to robotics [12, 30, 36] also have the advantage of potentially being hardware-agnostic.

There are many examples of computation-focused approaches to problems in robotics. Grasping, for example, can be posed as a contact point planning problem [36], which is appealing as it abstracts the problem away from the hand and the environmental context. Likewise, perception has been commonly seen as a passive, purely computational problem in which the robot has to process sensor information [33].

2) *Embodiment:* Tailoring the hardware to a particular problem can reduce the required computation. Hardware solutions are often simple and robust, especially when uncertainty plays a dominant role.

Grasp planning can benefit substantially from embodiment, as exemplified by simple under-actuated robotic hands [9, 14, 15, 35]. These hands exploit passive compliance to robustly grasp a variety of objects. Although this comes at the cost of reduced controllability, compliance removes the computational requirements of grasping while increasing grasp performance.

Appropriate embodiment also facilitates perception. For example, under-actuated hands reduce the need for accuracy in object pose estimation. Moreover, placing a vision sensor on the robot arm increases the sensor's field of view and reduces the effect of occlusions [1].

3) *Our Design Choice on the Spectrum:* We made deliberate design choices to facilitate computation. On the one hand, we reduced the need for computation by using an underactuated end-effector. The *reduced* number of degrees of freedom simplified grasp planning and object pose estimation. On the other hand, we reduced the need for computation by

*increasing* the number of degrees of freedom by mounting the robot arm on a mobile base. This allowed us to generate motion mostly through feedback control, rather than resorting to motion planning. However, in the Amazon Picking Challenge we failed to pick one challenge object (Sec. IV-C5, Fig. 8d) due to our chosen embodiment.

### C. Planning vs. Feedback

Classical robotics and AI employed the sense-plan-act paradigm, assuming the robot can build a perfect model of the world. In the 1980s, the difficulty of obtaining such models became apparent, initiating a shift towards feedback-driven approaches [8]. Interestingly, the control community shifted again into the opposite direction, from locally convergent controllers to global approaches, such as optimal control [5].

1) *Planning:* Planning finds global solutions, often with theoretical guarantees, where controllers based on local feedback would fail. The most common application of planning in robotic manipulation is motion planning [30]. To use these methods, practitioners have to provide models of the environment, calibrate the robot [21], and localize it in the environment [48]. Under these prerequisites, motion planners serve as general and versatile black-box solvers. Consequently, motion planning methods have been used successfully in many applications [29].

2) *Feedback:* If global search is not required or not possible, feedback control based on task-relevant features is often sufficient to generate successful robot motion. Visual servoing [18], for example, closes a feedback loop around feature motion, permitting robust achievement of manipulation tasks. Manipulation tasks, in particular, can be greatly simplified by exploiting feedback from contact with the environment [17, 31].

These feedback approaches are particularly useful in the presence of uncertainty, high dimensionality, long time horizons, and inaccurate models. In these cases, planning would be time-consuming, computationally demanding, and often intractable [38]. In contrast, feedback handles uncertainty and partial or imprecise world models by continuously incorporating local sensor information to adjust the executed motion.

3) *Our Design Choice on the Spectrum:* Our system relies on very simple planning. We use on-line grasp approach planning and execute the motions using pre-defined, feedback-guided motion primitives, thus avoiding configuration-space motion planning altogether (Sec. III-B). This positions our solution far to the feedback-side of the spectrum, standing in contrast to the majority of the other challenge entries (80% of the teams used motion planning, 44% used MoveIt [47], [13]). Feedback control is so successful in the Amazon Picking Challenge setting because the task only requires a limited range of motions, and the shelf provides plenty of contact surfaces to generate useful feedback. Recent work [28] demonstrates that reactive feedback without planning is sufficient in different realistic logistics settings as well. However, our evaluation (Sec. IV-C) shows that some shortcomings of our system, such

as the lack of in-bin reorientation of objects, must be addressed by some form of planning.

#### D. Generality vs. Assumptions

This spectrum is relevant for many problems in robotics and AI, for example in the strong vs. weak AI discussion, the no free lunch theorem [51], the bias-variance trade-off in machine learning [19], and the balance between generality and specificity in system design [50]. Finding completely general solutions is not possible [51] and we therefore must carefully select the “right” assumptions to build systems that are as general as possible in regards to the variant characteristics while making strong assumptions on those that are invariant.

1) *Generality*: Finding general solutions is, of course, an important goal in robotics. When we are able to solve not just one specific instance of a problem, but the problem in a general way, these general solutions reflect a deep understanding of the problem at hand. Additionally, the more general a solution is, the more useful it can potentially be. By definition, general solutions apply to a wide range of problems. In contrast, solutions strongly tailored to a specific problem instance (e.g. a robot demo) might not lead to insights or contribute to a broader understanding of system building.

There are a number of general approaches that were successfully applied in robotics. Task-generic planning algorithms such as  $A^*$  are widely used for mobile robot navigation. Recursive Bayesian estimation is a very generic framework that helped solving many different problems in robotics.

2) *Assumptions*: In machine learning, search, and optimization, the no free lunch theorems prove that no problem can be solved without making appropriate assumptions [51, 52]: averaged over all possible problems, there is no method that outperforms random guessing. The only way to improve on random guessing is by making assumptions about the problem.

We believe that problems in robotics are characterized by a significant amount of reoccurring underlying structure (for example, the laws of physics). Making suitable assumptions about this structure might enable many solutions that remain general over instances of robotic problems. To make progress towards effective solutions, we must therefore find suitable assumptions. We must understand and share knowledge of these assumptions to advance robotics.

Incorporating assumptions alleviates the difficulties of general purpose solutions. In motion planning, adding information about workspace connectivity can reduce the computational complexity by up to three orders of magnitude [42]. In reinforcement learning, adding explicit knowledge about physics makes the learning problem tractable by reducing its dimensionality [25, 44]. Similarly, the recent success of motor primitive learning can be largely attributed to dynamic movement primitives, which represent motion as a set of dynamical systems [23] with few parameters to learn [46], and thus provide suitable assumptions to restrict the space of robot motion.

3) *Our Design Choice on the Spectrum*: Our Amazon Picking Challenge system used available (general) solutions

whenever they proved sufficient to solve the problem. But when needed, we incorporated as much assumptions about the specifics of the competition as possible. Since we could not find existing approaches for reliably locating the target object, we used various assumptions to simplify the problem, e.g. that the objects are placed in a known shelf, which allows for the use of shelf related features or that the contents of each shelf bin are known and therefore only a small number of objects need to be considered (Sec.III-C). Our general solutions included hybrid automata to define the behavior of our system, a particle filter for localizing the robot, and standard joint-space and operational space controllers for motion generation.

## VI. CONCLUSION

We presented and evaluated our winning system for the 2015 Amazon Picking Challenge. To describe the system, we proposed four key aspects of system building. A systematic description of robotic systems according to these aspects (and additional ones proposed by others in the future) will facilitate accumulating general knowledge for building robotic systems. Each of the four aspects spans a spectrum of solutions.

Our entry to the Amazon Picking Challenge focused on certain regions of each spectrum. We observed that the placement of our system along these spectra differed significantly from other competition entries, indicating that these choices were important for our success. We propose that our four choices indicate suitable regions for problems in mobile manipulation, of which we view the Amazon Picking Challenge to be an instance.

First, as we have not yet developed a general solution to manipulation, we should emphasize tight integration between perception, planning, control, and hardware design, avoiding premature modularization (aspect A).

Second, manipulation requires contact with the environment. Modeling contact interactions reliably and efficiently is extremely difficult. This suggests a focus on embodiment, avoiding complex computation that might not adequately describe the real world (aspect B).

Third, manipulation planning is difficult due to high-dimensional configuration spaces, the effects of uncertainty, and long planning horizons. At the same time, sophisticated sensing capabilities are readily available. This suggests to exploit feedback wherever possible (aspect C).

Finally, we should acknowledge that assumptions are necessary to solve complex problems. Finding appropriate assumptions therefore is an important goal of robotics research. To find these assumptions, we advocate solving concrete problems with strong assumptions, relaxing them incrementally (aspect D).

Our lessons on building robotic systems are consistent with those others derived from their experience in similar robotics challenges. In the area of autonomous driving [10] and humanoid robotics [2], the resulting insights have led to significant advances. We hope that the Amazon Picking Challenge and our lessons learned will be equally useful for manipulation.



## REFERENCES

- [1] J. Aloimonos, I. Weiss, and A. Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1(4): 333–356, 1988.
- [2] C. G. Atkeson, B. P. W. Babu, N. Banerjee, D. Berenson, C. P. Bove, X. Cui, M. DeDonato, R. Du, S. Feng, P. Franklin, et al. No falls, no resets: Reliable humanoid behavior in the DARPA robotics challenge. In *IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 623–630. IEEE, 2015.
- [3] C. Y. Baldwin and K. B. Clark. *Design rules: The power of modularity*, volume 1. MIT press, 2000.
- [4] Barrett. Barrett WAM. <http://barrett.com/products-arm.htm>, 2015. Accessed: 2016-01-29.
- [5] D. P. Bertsekas. *Dynamic programming and optimal control*. Athena Scientific Belmont, MA, 1995.
- [6] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. Sukhatme. Interactive perception: Leveraging action in perception and perception in action. *IEEE Transactions on Robotics*, 2016. URL <http://arxiv.org/abs/1604.03670>. Under review.
- [7] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.
- [8] R. A. Brooks. Elephants don't play chess. *Robotics and Autonomous Systems*, 6(1):3–15, 1990.
- [9] E. Brown, N. Rodenberg, J. Amend, A. Mozeika, E. Steltz, M. R. Zakin, H. Lipson, and H. M. Jaeger. Universal robotic gripper based on the jamming of granular material. *Proceedings of the National Academy of Sciences*, 107(44):18809–18814, 2010.
- [10] M. Buehler, K. Iagnemma, and S. Singh. *The 2005 DARPA grand challenge: the great robot race*, volume 36. Springer Science & Business Media, 2007.
- [11] P. R. Cohen. Empirical methods for artificial intelligence. *IEEE Intelligent Systems*, 11(6):88, 1996.
- [12] D. Coleman. MoveIt! strengths, weaknesses, and developer insight. Presentation at ROSCon, 2015.
- [13] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman. Lessons from the Amazon Picking Challenge. *ArXiv e-prints*, January 2016.
- [14] R. Deimel and O. Brock. A novel type of compliant and underactuated robotic hand for dexterous grasping. *The International Journal of Robotics Research*, 35:161–185, 2016.
- [15] A. M. Dollar and R. D. Howe. The highly adaptive SDM hand: Design and performance evaluation. *The International Journal of Robotics Research*, 29(5):585–597, 2010.
- [16] M. Egerstedt. Behavior based robotics using hybrid automata. In *Hybrid Systems: computation and control*, pages 103–116. Springer, 2000.
- [17] C. Eppner, R. Deimel, J. Álvarez Ruiz, M. Maertens, and O. Brock. Exploitation of environmental constraints in human and robotic grasping. *The International Journal of Robotics Research*, 34(7):1021–1038, 2015.
- [18] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, 8(3):313–326, 1992.
- [19] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2): 83–85, 2005.
- [20] N. Hawes, M. Zillich, and P. Jensfelt. Lessons learnt from scenario-based integration. In *Cognitive Systems*, pages 423–438. Springer, 2010.
- [21] J. Hollerbach, W. Khalil, and M. Gautier. Model identification. In B. Siciliano and O. Khatib, editors, *Springer Handbook of Robotics*, pages 321–344. Springer, 2008.
- [22] R. Holmberg and O. Khatib. Development and control of a holonomic mobile robot for mobile manipulation tasks. *The International Journal of Robotics Research*, 19(11): 1066–1074, 2000.
- [23] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal. Dynamical movement primitives: learning attractor models for motor behaviors. *Neural Computation*, 25(2):328–373, 2013.
- [24] M. Johnson, B. Shrewsbury, S. Bertrand, T. Wu, D. Duran, M. Floyd, P. Abeles, D. Stephen, N. Mertins, A. Lesman, et al. Team IHMC's lessons learned from the DARPA robotics challenge trials. *Journal of Field Robotics*, 32(2):192–208, 2015.
- [25] R. Jonschkowski and O. Brock. Learning state representations with robotic priors. *Autonomous Robots*, 39(3): 407–428, 2015.
- [26] R. Jonschkowski, C. Eppner, S. Höfer, R. Martín-Martín, and O. Brock. Probabilistic multi-class segmentation for the Amazon Picking Challenge. Technical Report RBO-2016-01, Robotics and Biology Laboratory, Department of Computer Engineering and Microelectronics, Technische Universität Berlin, February 2016.
- [27] D. Katz and O. Brock. A factorization approach to manipulation in unstructured environments. In *Robotics Research—The 14th International Symposium ISRR (2009)*, pages 285–300. Springer, 2011.
- [28] R. Krug, T. Stoyanov, V. Tincani, H. Andreasson, R. Mosberger, G. Fantoni, and A. J. Lilienthal. The next step in robot commissioning: Autonomous picking & palletizing. *IEEE Robotics and Automation Letters*, 2016.
- [29] J. P. Laumond. Kineo CAM: a success story of motion planning algorithms. *Robotics & Automation Magazine, IEEE*, 13(2):90–93, 2006.
- [30] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, New York, NY, USA, 2006.
- [31] T. Lozano-Perez, M. T. Mason, and R. H. Taylor. Automatic synthesis of fine-motion strategies for robots. *The International Journal of Robotics Research*, 3(1):3–24, 1984.
- [32] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige. The office marathon: Robust navigation

- in an indoor office environment. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 300–307, May 2010.
- [33] D. Marr. *Vision: A computational approach*. Freeman & Co., San Francisco, 1982.
- [34] R. Martín-Martín and O. Brock. Online interactive perception of articulated objects with multi-level recursive estimation based on task-specific priors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2494–2501. IEEE, 2014.
- [35] M. T. Mason, A. Rodriguez, S. S. Srinivasa, and A. S. Vazquez. Autonomous manipulation with a general-purpose simple hand. *The International Journal of Robotics Research*, 31(5):688–703, 2012.
- [36] A. T. Miller and P. K. Allen. GraspIt! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 11(4):110–122, 2004.
- [37] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal. Operational space control: A theoretical and empirical comparison. *The International Journal of Robotics Research*, 27(6):737–757, 2008.
- [38] C. H. Papadimitriou and J. N. Tsitsiklis. The complexity of Markov decision processes. *Mathematics of operations research*, 12(3):441–450, 1987.
- [39] R. Pfeifer and G. Gómez. Morphological computation—connecting body, brain, and environment. In *Creating Brain-Like Intelligence*, pages 66–83. Springer, 2009.
- [40] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. ROS: an open-source robot operating system. In *ICRA workshop on open source software*, 2009.
- [41] C. Rennie, R. Shome, K. E. Bekris, and A. F. De Souza. A Dataset for Improved RGBD-based Object Detection and Pose Estimation for Warehouse Pick-and-Place. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.
- [42] M. Rickert, A. Sieverling, and O. Brock. Balancing exploration and exploitation in sampling-based motion planning. *IEEE Transactions on Robotics*, 30(6):1305–1317, 2014.
- [43] R. B. Rusu and S. Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, 2011.
- [44] J. Scholz, M. Levihn, C. Isbell, and D. Wingate. A physics-based model prior for object-oriented MDPs. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1089–1097, 2014.
- [45] K. Schwaber. *Agile project management with Scrum*. Microsoft Press, 2004.
- [46] F. Stulp and O. Sigaud. Robot skill learning: From reinforcement learning to evolution strategies. *Paladyn, Journal of Behavioral Robotics*, 4(1):49–61, 2013.
- [47] I. I. Sucas and S. Chitta. MoveIt. <http://moveit.ros.org>, 2016.
- [48] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. MIT Press, 2005.
- [49] K. Ulrich. The role of product architecture in the manufacturing firm. *Research policy*, 24(3):419–440, 1995.
- [50] J. P. van Gigch. *System Design Modeling and Metamodeling*. Springer Science & Business Media, 1991.
- [51] D. H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390, 1996.
- [52] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.
- [53] XR4000. Nomadic technologies inc. XR4000. [http://cbcis.ttu.edu/ep/old\\_netra\\_site/about/xr4000/XR4000.htm](http://cbcis.ttu.edu/ep/old_netra_site/about/xr4000/XR4000.htm), 2015. Accessed: 2016-01-29.