Vision-less Bin-Picking for Small Parts Feeding

Mathiesen, Simon; Iturrate, Iñigo; Kramberger, Aljaz

Published in:
2019 IEEE 15th International Conference on Automation Science and Engineering, CASE 2019

DOI:
10.1109/COASE.2019.8843203

Publication date:
2019

Document version:
Accepted manuscript

Citation for pulished version (APA):
Mathiesen, S., Iturrate, I., & Kramberger, A. (2019). Vision-less Bin-Picking for Small Parts Feeding. In *2019 IEEE 15th International Conference on Automation Science and Engineering, CASE 2019* (pp. 1657-1663). IEEE. https://doi.org/10.1109/COASE.2019.8843203

Vision-less Bin-Picking for Small Parts Feeding

Simon Mathiesen, Iñigo Iturrate and Aljaž Kramberger
SDU Robotics, The Maersk McKinney Moller Institute
University of Southern Denmark, 5230 Odense M, Denmark
Email: {simat, inju, alk} @mmmi.sdu.dk

***Abstract*— Part feeding remains a vital yet troublesome task. This is especially true for flexible manufacturing systems, where classical methods are less applicable due to constraints from small batch sizes and high part variance. In this paper, a novel approach to part feeding is presented, which is targeted at small parts, such as nuts and bolts, that commonly occur in assembly tasks. The approach is an alternative bin picking solution, where parts stored unordered in bins, are oriented and presented for further manipulation using a specially designed tool. The tool is manifested a scoop, which a robot can use to scoop parts from the bins without the use of external sensory input such as vision. Using solely mechanical orienting principles and the motion of the robot, the parts are sorted so only parts in a desired orientation remain in the scoop. The approach is tested on a number of different parts to benchmark its performance.**

I. INTRODUCTION

The application of industrial automation is expanding into new areas as it is no longer only relevant for mass production. When it comes to small batch production equipment costs must be kept low or the equipment have the ability to be reconfigured, and reused across multiple production tasks. Furthermore, some tasks are either too complex or need to be performed so infrequently that they are not worth automating and thus production with human-robot collaboration can be the solution. Regrettably, what is easy for humans is not necessarily easy to implement with robots.

One recurring task is part feeding, i.e. the task of introducing parts, typically from bulk, into the robotic workcell. Albeit troublesome, it is often beneficial to store parts in bulk, even in pure robotic workcells, as it is both cheap and space efficient. Classical feeding solutions for handling bulk parts, such as vibratory bowl feeders, can be made highly efficient, but they are generally less suited for a flexible robotic workcell, which must handle small batch production with high part variance. This is due to the fact that significant work goes into optimizing the feeder to a specific part. Although there exists a range of flexible part feeders more suited for this type of production, they generally suffer from other drawbacks, such as increased complexity, reduced feed rate, and high equipment cost. Therefore, alternative efficient solutions for part feeding are of general interest.

This work presents a new approach for feeding of small parts like nuts, bolts, washers and the like, which occur frequently in industrial assembly tasks. The approach is grounded in classical part feeding, but is manifested as a tool for a robotic manipulator, rather than a separate device, and uses the mechanical features of the parts to reject those that are not in the desired orientation from the tool. The tool, shown in Fig. 1, resembles a *scoop* that can be used to shovel parts from bulk in a container. It was developed in connection with the Industrial Assembly Challenge for the World Robot Summit 2018[1]. The Assembly Challenge was centered around a belt drive unit. In one subtask the goal was kitting up the parts, which were then needed in the assembly. The majority of the parts to be used for the kitting were small light metal parts located in bins, as shown in Fig. 1, from which a human could easily pick them. The parts needed to be separated and the right amount of parts placed in specific compartments of a kitting tray, some in specific orientations.

Fig. 1: The tool, or *scoop*, will scoop up parts from the bins and shake them until only parts in a desired orientation remain in it. Parts can then be picked from it. Thereafter, the scoop can be returned to the rack and the robot can attach another[2].

The paper is structured as follows: Section II will describe the relevant state of the art centered around part feeding. Section III and section IV will explain the developed approach in detail, where section III will focus on the tool and section IV will explain how it is controlled to obtain the desired behavior. Section V will present results from experiments with the goal of quantifying the efficiency of the approach. Finally, in section VI we will elaborate on the results as well as the extendability and limitations of the approach, with conclusions given in section VII.

---

[1] Source: http://worldrobotsummit.org/en/wrc2018/

[2] An example execution can found at: https://youtu.be/Vi2bZ4rgxxk

II. STATE OF THE ART

The approach to part feeding presented in this paper is inspired by previous work on the topic of classical part feeding, such as Mathiesen et al. [1]. This work was centered around the vibratory bowl feeder and presented an approach to optimizing the geometries of mechanical orienting devices to fit the parts and ensure these are oriented as desired when leaving the feeder. This concept of orienting parts by exploiting their geometric features and physical properties can be referred to as classical or traditional part feeding and is also the core concept of the approach presented in this paper. The most comprehensive work on this topic was done by Boothroyd [2]. Amongst other topics, this book provides an extensive overview of the subject of vibratory part feeding. In addition to describing the dynamics of this feeder type, the book also provides an appendix, which assists a feeder designer in mapping parts to relevant mechanical orienting devices based on center of gravity and geometric features such as holes, chamfers, protrusions, etc.

Classical feeding techniques, such as the vibratory bowl feeder, are inherently inflexible as they are designed specifically to feed a single type of part. Some approaches have been made to remedy this, for example in Redford et al. [3]. Here, they propose a linear vibratory feeder system similar in concept to the widely used bowl design. The entire linear track – which ensures correct part orientation – is replaceable, thus enabling the reuse of the feeder across multiple tasks. To increase the flexibility of the vibratory bowl feeder Joneja and Lee [4] proposed a modular design, where the orienting devices of the bowl could be replaced and adjusted, thereby making it possible to reconfigure the feeder to handle other parts. In principle, both approaches add some much-needed flexibility to some otherwise very dedicated systems, but are, to the best of our knowledge, not widely used in the industry for various reasons, including the need of involving skilled personnel in the reconfiguration process. Other drawbacks of the approaches mentioned so far are that these are separate devices accounting for a significant footprint in a workcell and that they are inherently not capable of feeding more than one part type at a time per feeder. Even the more modern flexible part feeders, such as the *anyfeed*[3] or the *Asycube*[4] suffer from these drawbacks, with the addition of a significant price tag and the added complexity of the necessary computer vision solution for finding parts to pick. Running out of workable workspace due to problems with clutter and reach is especially evident for assembly tasks that consist of many different parts. Keeping the feeding solution small and inside the cell should, therefore, be pursued in flexible manufacturing, and utilizing the advanced manipulation skills of the robot to this end seems obvious. One such approach was explored in Goldberg [5], which presented a method for aligning parts to specific orientations without sensory input. An algorithm planned sequences of squeezing actions to be carried out by a parallel gripper, ensuring object alignment. However, the approach was limited to planar parts and furthermore required parts to be singulated prior to the aligning process.

More recently, Bin Picking [6] has been introduced to the industry as a part feeding technology that typically makes use of computer vision to locate parts in a bin and then plan trajectories that pick them up. Picking can be done directly from the desired orientation, or arbitrarily, such that a second step is required to reorient the part [7]. Due to the complexity of estimating parts poses and orientations in the first case, or to the time needed to perform the pickup-reorientation sequence in the second case, bin-picking solutions are typically slow. Cycle times can fall in the range of $\sim$ 19 seconds when using a planner that returns optimized picking trajectories [6], but potentially of up to $\sim$ 115 to 165 seconds for a non-optimized cycle of picking and reorientation [7].

III. TOOL DESIGN FOR VISION-LESS BIN-PICKING

The proposed approach to part feeding can, in short, be described as scooping up parts from bins with the developed tool mounted at the end-effector of the robot, where this tool afterward functions as a fixture in which parts are essentially kitted up for further manipulation. The generalized version of the tool is therefore referred to as a *Scoop*. Scoops are designed to work for a specific part, or range of parts, that share similar features and dimensions. They utilize the same concept of mechanical orientation techniques as a vibratory bowl feeder, namely, exploiting the geometric features and mass-properties of the part to orient it.

A. Part Orientation from Geometric Features and Mass Properties

The general principle of mechanical part orientation relies on the parts being in motion. Traditionally, the motion is produced using directed high-frequency vibrations that make parts convey smoothly in a certain direction. Typically parts move on and along a track with a geometry that aims to ensure that they move in a single file. While in motion, they encounter changes to the track, such as protrusions, grooves, narrowing or drops, that either result in parts in undesired orientations falling off the track and getting rejected, or them getting reoriented as desired. The major difference in this approach compared to traditional feeders is that the motion required to orient parts is produced by the robot holding the scoop, thus eliminating the need for a separate device. This method of part feeding is in principle directly transferable from vibratory bowl feeder design and shares many of its constraints. The core principles and constraints that make up a *Scoop* are listed below:

- Parts must be small enough to be contained within the scoop.
- The geometry of the scoop should guide parts into a groove to minimize rejections.
- Parts can only be fully oriented if they have non-equal dimensions or specific features.

---

[3] Source: https://www.flexfactory.com

[4] Source: https://www.asyril.com

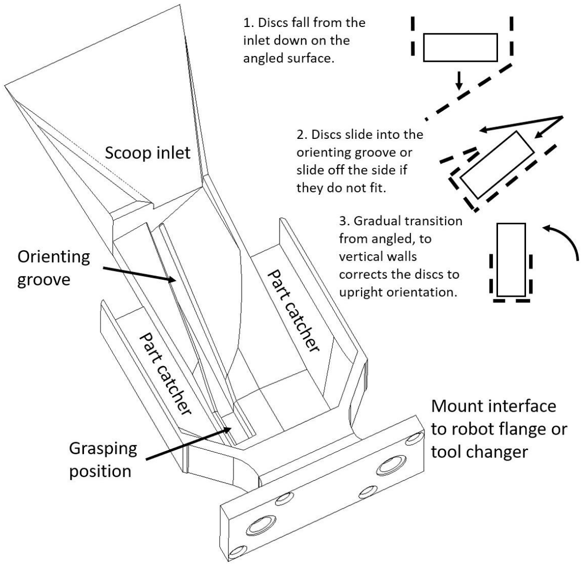


Fig. 2: Illustration of the anatomy of the general scoop with the orienting principle described for disc-shaped objects.

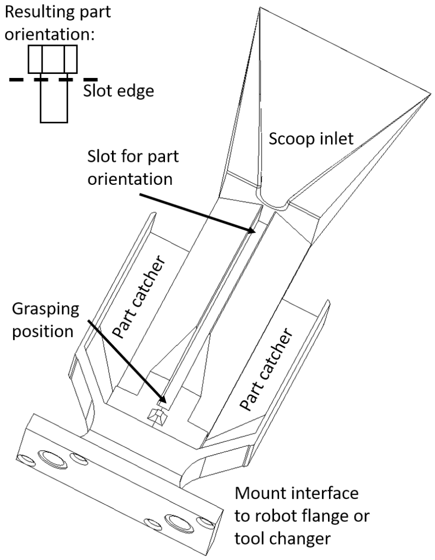


Fig. 3: Illustration of the modified scoop designed specifically for feeding bolts.

- The orientations of the parts are maintained as they enter groove.
- The geometry of the scoop should be designed so that shaking the scoop causes:
  - ... parts in undesired orientations to fall off the scoop.
  - ... parts in the desired orientation to remain in the scoop.
- Parts need to protrude enough from the scoop to be possible to pick.

1) The General Anatomy of a Scoop: The general anatomy of the scoop and the principle of how it orients parts are shown in Fig. 2. Parts are scooped onto the *inlet* portion of the device from where they next *fall* into the *orienting groove*. The groove, together with the motion of the robot, ensures that parts not oriented as desired are discarded by sliding off the sides of the scoop and into the *part catchers* or back into the bin. The design of the groove (effectively the width) should ensure that no two parts can be located side by side and thus the total number of parts that can be in the groove at any time is determined by its length. This effectively singulates the parts. Parts caught in the *part catchers* can be poured back into the bin by tilting the scoop forward.

2) Orienting Discs: The general design described above was used for disc-like parts, such as washers, nuts, and end-caps used in the WRS Industrial Challenge.

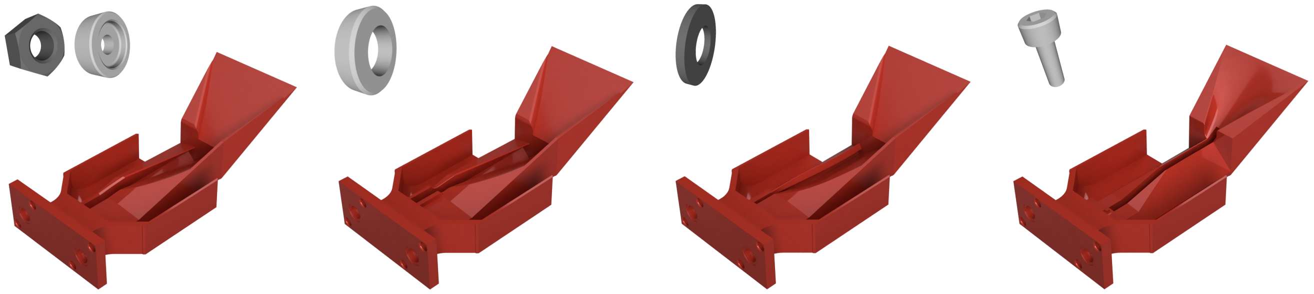
The main design parameters for this specific scoop type are the opening *width* of the inlet, and the *width* and *height* of the groove. Furthermore, the rate at which the walls of the orienting groove should transition from an angled to an upright orientation is determined by the diameter of the disc. If the disc is large the rate of transition must decrease accordingly to avoid jamming, thus making the required length of the groove longer.

3) Orienting Bolts: We found that the scoop design used for disc-like objects did not translate well to orienting and feeding bolts, which are characterized by having a cylindrical main body with a larger diameter head protruding from one end. However, the design of the scoop can be adapted to exploit the geometrical features of the bolt for orientation purposes as described in Boothroyd [2]. Fig. 3 shows the modified scoop design for feeding bolts. The main body of the bolts is intended to slide into the centralized groove and be caught on the protruding head, as shown in the top left corner of Fig. 3. The groove is made up of what is essentially two right-angled triangles with their hypotenuses facing away from the center. Thus, bolts that are not funneled into the groove slide off the sides and into the bin or *part catchers*.

The main design parameter of this scoop is the width of the groove, and this should be wider than the main body of the bolt, but narrower than the diameter of its head.

B. Flexibility from reconfigurability

For the process to be flexible when it comes to handling various parts, the robot needs to be able to automatically exchange the tool for different scoops when a new part is needed in the production system. In analogy to the well-known tool exchanger, where a pin locks the tool to the stationary part of the tool exchanger mounted at the Tool Center Point (TCP) of the robot, we use a smaller, pneumatically actuated version with similar properties. The scoop exchange system is mounted on the robot's gripper, providing better usability during the manufacturing process. To ensure efficient reconfiguration, all scoops have the same mounting interface, consisting of four pilot holes for aligning the scoop with the exchange system and two slightly bigger holes for fitting the locking pins (Figs. 2 and 3). This



(a) This scoop design is used for both nuts and end-caps.

(b) This scoop design is similar to the one for nuts and end-caps, but has a wider groove and higher walls for containing the larger spacer ring.

(c) This scoop design is used for washers and thus has a significantly narrower groove.

(d) This scoop design is designed for bolts in the M3-size.

Fig. 4: The four scoop designs and their associated parts used for validating scooping as an approach for feeding small parts.

mechanism provides a stiff and reliable connection between the exchangeable scoop and the stationary robot gripper.

A specialized scoop was designed for each of the parts as shown in Fig. 4 and stored on a scoop holder (Fig. 1). The function of the holder is to store the scoops at pre-programmed positions in the workcell. When a new scoop is needed, the robot approaches the holder and places the current scoop at its storage location. Afterward, the robot moves to the location of the new scoop, then attaches it and lifts it out of the holder. As a result of this, the robot is reconfigured to facilitate feeding operation of a different part.

IV. ROBOT MOTION CONTROL

In order to fully orient the parts within the scoop, the robot must go through a series of motions that help remove excess parts from the scoop inlet, remove incorrectly oriented parts that have already fallen into the orientation groove, and guide parts that are already in the orientation groove into the grasping position at its rear end. For this purpose, we make use of hybrid position and force control. Our control is defined as a series of small, self-complete primitives that are then sequenced together through a series of input transition conditions given by the force-torque measurement at the robot tool.

A. Force-Controlled Primitives

Defining a force-controlled task as a sequence of primitives is not a new concept, and can be traced back to as early as Mason et al. [8] and Morrow et al. [9]. More recent work includes Thomas et al. [10]. Essentially, these approaches all make use of a similar parametrization of a force-controlled robot movement, as described in Mason et al. [8], where a task is formalized in terms of models for the manipulator, task constraints (including reference frames), and control strategy. Primitive-based robot programming is also common in the field of programming by demonstration as a representation of a learned task, and our set of primitives described below is comparable to those used by Stenmark et al. [11].

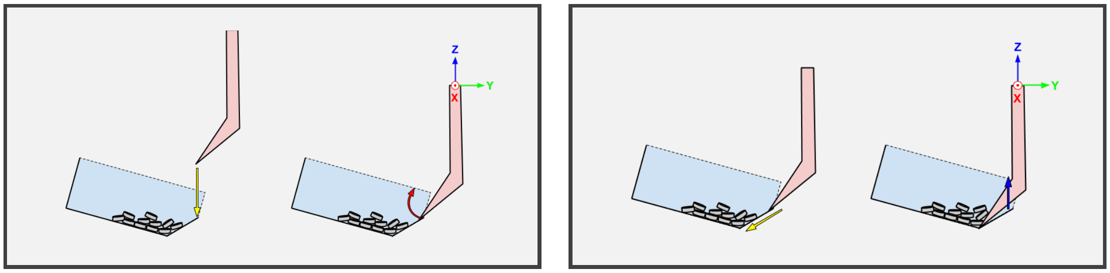
In this work, our part-orientation sequence is defined from a set consisting of the following primitives:

- *Move/rotate until force/torque contact:* The robot moves or rotates along the specified axis until an input force or torque of a certain magnitude is measured at the robot end-effector. The movement can be either in Cartesian space or in robot joint space.
- *Move/rotate compliantly while applying force/torque:* This primitive can be wrapped around a position-controlled motion. The robot will use admittance control along the specified axis while tracking the input force/torque setpoint. This can be used to e.g. push along a specified axis with a target force while moving along a different axis.
- *Move/rotate for distance:* The robot moves or rotates for the specified distance or angle.

For our purposes, we only make use of the limited functionality described in the above list. Note, however, that this set of primitives could be generalized to fall under the previously-cited schemes.

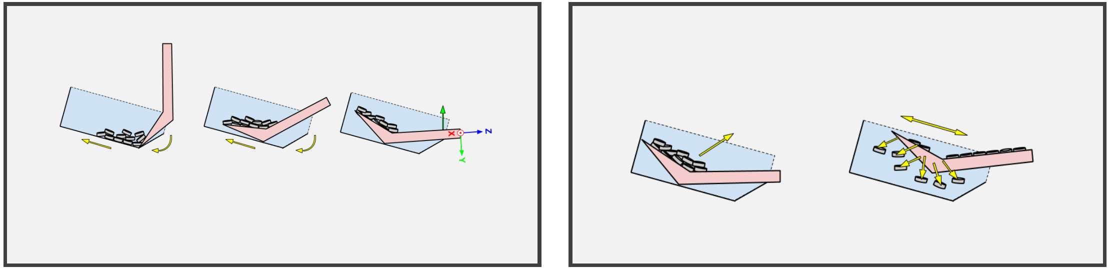
B. Programming Part-scooping Motions

Fig. 5 shows the sequence of force-controlled primitives used for our vision-less bin-picking strategy. The motions can be decomposed into three main stages: In the first stage (Fig. 5a-5c), the robot performs a scooping motion that fills the *Scoop* with parts from the bin. In the second stage (Fig. 5d), the robot performs a series of rotations and back-and-forth and sideways movements at high speed and acceleration to make the parts impact with the orienting geometry of the *Scoop* and thus remove the parts that are not in the desired position and orientation in the orientation groove. Finally (Fig. 5e), the robot angles the scoop forward to remove parts that may remain in the part catcher, while keeping those in the scoop groove. It then retracts into a picking position.



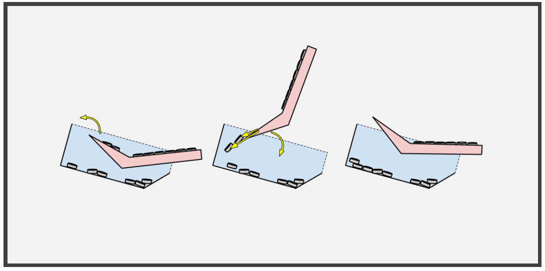
(a) In the first stage, the part-scooping tool is moved down vertically until contact with the edge of the parts bin. This results in a torque around the end-effector x-axis

(b) The scoop is slid down compliantly along the entrance slope of the bin by applying forces in the negative z-axis and negative y-axis, until a large contact force is detected along the robot tool flange z-axis.



(c) The robot is moved compliantly in a scooping motion by applying forces along the end-effector negative z-axis and positive y-axis and a torque around the x-axis, until a large force is detected along the negative y-axis.

(d) Purely in position control, the scooping tool is retracted to a safe distance from the bin. It is then shaken for a travel distance of 1.5 cm in a sequence of ten sideways movements and ten back-and-forth movements, with a velocity of 2 m/s and an acceleration of 6 m/s^2 for all parts except the washers, which are shaken with 1 m/s velocity and 5 m/s^2 acceleration. This shaking sequence is repeated three times.



(e) The scooping tool is then rotated forward around the tool x-axis in order to make sure that parts in the entrance slope are removed by sliding down. It is then rotated back and moved to the part pick-up position.

Fig. 5: Sequence of force-controlled primitives used for vision-less scooping and shaking of small parts. Arrows in yellow represent movement. Other-colored arrows represent forces (straight) and torques (curved) measured by the robot on the tool (shown here as a coordinate axis on the scoop mount side), and used as transition conditions between movement primitives.

V. EXPERIMENTAL EVALUATION

We evaluated several scoop designs in real-world experiments. The experimental setup consisted of

- Universal robot UR10e robot arm. The robot arm has an embedded force/torque sensor at the TCP, enabling force measurement and tracking.
- Weiss WSG 50 servo controlled gripper.
- Gripper fingertip exchange system, providing a rigid mechanical coupling and automatic exchange of scoops for different parts.

The different scoop designs were evaluated on small parts commonly found in production facilities, namely: *Hex M3 bolts*, *M3 nuts* and *washers*, *spacing rings*, and their corresponding *end caps*. All of these are shown in Fig. 4. The parts were placed in part bins, which were located on a specially designed rack, enabling the robot to reach the bins and perform the scooping motion. The experiments were executed in several steps. In the first step, the robot picked the scoop from the scoop holder and moved to the required part bin. Secondly, the scooping motion was executed as described in Sec. IV. In the third step, the scooped parts were aligned with the specially programmed shaking motion, and, in the final step, the parts aligned in the groove were evaluated and the rest emptied back into the bin.

Through the course of the experiment we statistically evaluated the outcome based on the following parameters:

- Number of parts scooped before shaking.
- Number of parts successfully aligned after shacking and emptying the scoop.
- Number of parts aligned per minute.

In total 50 scooping motions per part were executed and evaluated by a human observer. If a part was scooped properly and misaligned after the shaking motion, it counted as a failed experiment. The results are shown in Fig. 6 and Table I. Smaller parts, e.g. nuts, bolts, and washers were naturally easier to scoop out of the bin, whereas the scoop due to its size could not fit as many of the larger spacer rings. The comparably low score of the end-caps was due to a lack of parts in the bin (we only had 7, compared to 80+ for the nuts). With more parts it is expected to reach the same score as for the nuts as they are comparable in size.

The second evaluated parameter described how well the parts aligned in the orientation groove of the scoop. This step is more relevant than the scooping itself because the subsequent task relies on the accurate positioning of the part. If the part is misaligned at the grasping position in the scoop, grasping of the object will be imprecise or even fail. Fig. 7 and Table I shows that, in general, more than two parts successfully align in the groove, thus giving a high chance of successfully executing any subsequent actions. In some cases we observed the parts getting stuck together when the robot executed the shaking motion. This led to one or more parts not properly aligning in the scoop. However, statistics show that, on average, more than one part was scooped, so a single misaligned part does not compromise the overall success of the scooping action.

The last evaluation parameter represents a statistical estimate of the successfully aligned parts per minute, also referred to as *feed rate*. All of the scooping actions were carried out with the same time interval of $35s$, as the robot motion is entirely pre-planned and deterministic. The normalized results can be seen in Table I.

	Nut	M3 Bolt	Washer	Spacing ring	End cap
Avg. of successful parts scooped	18.6	13	11.7	5.6	3.3
Avg. of successful aligned parts	6.2	2.8	3.4	2.9	2.1
Likelihood of the part aligning [%]	100	99.3	94	89.1	100
Number of parts aligned per minute	10.5	4.8	5.7	5.0	3.5

TABLE I: Statistical evaluation of robot scooping parts with 50 repetitions of the task.

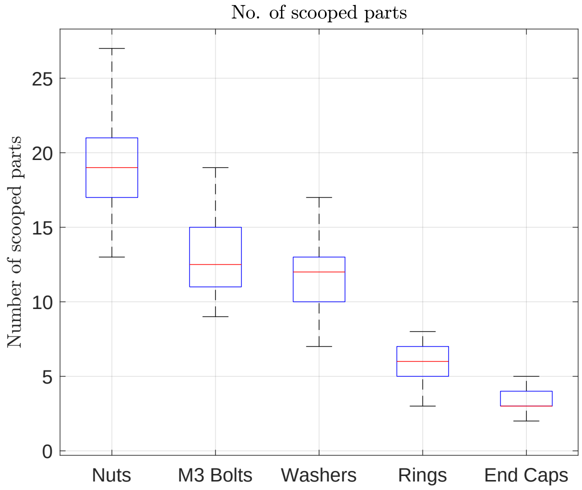


Fig. 6: Number of parts scooped out of the part bin.

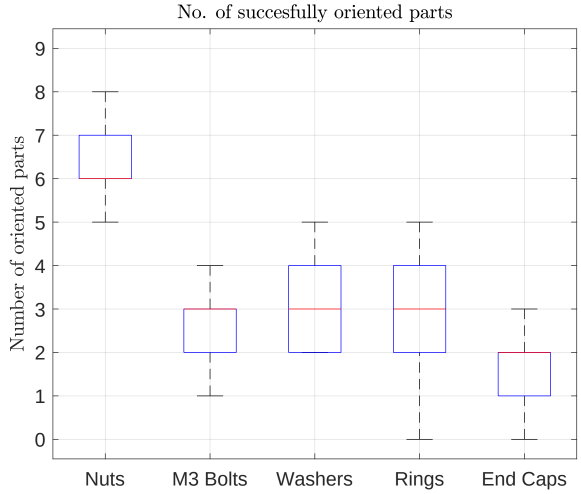


Fig. 7: Number of successfully oriented objects in the appropriate position for subsequent actions.

VI. DISCUSSION

Initially, the concept of *scooping*, as described here, was developed for a kitting task where a few of each type of part were to be placed in specific compartments of a tray. For this task, the solution was quite suitable, as if more parts than needed were scooped, they could be stored for further use by placing the scoop back in its holder. This added a valuable level of flexibility to the approach.

For tasks where a large number of the same part type are needed, it is clear from the results in Sec. V that the approach in its current state will not be able to compete with highly optimized feeding solutions with feed rates of multiple parts per second. However, from a systems perspective, the requirements on speed are not as high for flexible automation as they are for hard automation doing mass production. The number of parts fed per minute with our feeding solution is likely sufficient for a broad range of applications. That said, there is potential for improvements to both the robot motion and the design of the scoops when it comes to achieving higher feed rates if we allow for part-specific variations, as the approach covered in this work emphasized generality across parts. Concerning the robot motion, this was already evident for washers, which required a reduction of speed and acceleration to produce consistent performance. Similarly, adapting the motion of the robot for other parts could help keep more parts in the scoop, thus increasing an already considerably high efficiency. Concerning the scoop design itself, many different improvements are possible.

So far, the scoops have been manifested in two design concepts: One for disc-shaped parts and one for bolts. The generic design presented in Fig. 2 was initially tested for bolts, but we found that it was not sufficient for this specific part type as the bolts tended to get stuck in the groove in an undesired orientation. This problem was remedied by the design presented in Fig. 3.

In the experiments, other issues were observed related to unoptimized scoop designs. Some of these problems can easily be fixed, e.g. the spacer rings getting stuck in the part catchers, which were simply too narrow for that part. Another observed problem was the washers getting stuck on each other in the groove. Solving this would likely require more precise parameter optimization. Finally, we observed a single bolt that was initially properly aligned jump up from the groove when the scoop was rotated backward and moved to the pickup position. Although this could likely be solved by tuning the motion, there is a general need for further maturing of the scoop designs and very likely also for adapting the design to other part types in the same way as was done for the bolts. This would allow us to utilize additional features of the parts such as holes and protrusions to be able to feed, and fully orient, more complex geometries.

Furthermore, tuning the parameters of the scoop to fit the specific part is not always trivial, and thus the scoop design would benefit significantly from an automated parameter optimization approach. Our previous work [1] suggests that using simulation-based optimization is a viable approach worth exploring in the future. An automatic design approach could potentially aid in the identification of cases where a single scoop could be used to feed multiple parts, as was the case for the nuts and end-caps in this work.

Additionally, if future work proves that it is not possible to entirely prevent wrongly oriented parts from remaining in the scoop, an embedded sensor solution would allow for error handling by detecting cases where part-alignment completely fails and no parts end up in the desired orientation.

VII. CONCLUSION

This paper has presented a novel approach for feeding small parts stored unordered in bins. Part-feeding is achieved by using a specialized robot tool that first scoops the parts out of the bin and then – based on its design features and a preprogrammed robot motion – sorts the parts such that only those in the desired orientation remain in the scoop. Our results show that the efficiency of this approach varies with each scoop design and its associated part. In our experiments, the method demonstrated a part feed rate that is suitable for small-batch industrial assembly processes where manufacturing adaptability and flexibility are vital.

Naturally, there is room for improvement of the presented scoop designs as well as a need for further exploration of the part range for which the approach as a whole is applicable. Investigating the mentioned options, together with developing an automated process for designing the scoops directly from part CAD-data, are in our opinion the natural next steps for proceeding with what we believe is a promising alternative for small parts feeding for flexible industrial manufacturing.

ACKNOWLEDGMENT

The work was carried out as a part of SDU Robotics' participation in the World Robot Challenge 2018 / Industrial Robotics Category, an activity funded by SDU's I4.0 Lab.

REFERENCES

[1] S. Mathiesen, L. C. Sørensen, D. Kraft, and L.-P. Ellekilde, "Optimisation of trap design for vibratory bowl feeders," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3467–3474, 2018.

[2] G. Boothroyd, *Assembly Automation and Product design*, 2nd ed. CRC Press, 2005.

[3] A. Redford, E. Lo, P. Killeen, and A. Chisholm, "Parts presentation to multi-arm assembly robots," *CIRP Annals-Manufacturing Technology*, vol. 32, no. 1, pp. 399–404, 1983.

[4] A. Joneja and N. Lee, "A modular, parametric vibratory feeder: a case study for flexible assembly tools for mass customization," *IIE transactions*, vol. 30, no. 10, pp. 923–931, 1998.

[5] K. Y. Goldberg, "Orienting polygonal parts without sensors," *Algorithmica*, vol. 10, no. 2, pp. 201–225, 1993.

[6] L.-P. Ellekilde and H. G. Petersen, "Motion planning efficient trajectories for industrial bin-picking," *The International Journal of Robotics Research*, vol. 32, no. 9-10, pp. 991–1004, 2013.

[7] S. Koo, G. Ficht, G. M. García, D. Pavlichenko, M. Raak, and S. Behnke, "Robolink feeder: Reconfigurable bin-picking and feeding with a lightweight cable-driven manipulator," *13th IEEE Conference on Automation Science and Engineering (CASE)*, pp. 41–48, 2017.

[8] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 11, no. 6, pp. 418–432, 1981.

[9] J. D. Morrow, B. J. Nelson, and P. K. Khosla, "Vision and force driven sensorimotor primitives for robotic assembly skills," *IEEE/RSJ International Conference on Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots'*, vol. 3, pp. 234–240, 1995.

[10] U. Thomas, F. M. Wahl, J. Maass, and J. Hesselbach, "Towards a new concept of robot programming in high speed assembly applications," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).*, pp. 3827–3833, 2005.

[11] M. Stenmark and E. A. Topp, "From demonstrations to skills for high-level programming of industrial robots," *AAAI Fall Symposium Series: Artificial Intelligence for Human-Robot Interaction; AAAI: Arlington, VA, USA*, pp. 75–78, 2016.