Planning and optimization of robotic pick-and-place operations in highly constrained industrial environments

Bence Tipary[1,2], András Kovács[1] and Gábor Erdős[1,2]

[1] Institute for Computer Science and Control (SZTAKI), Eötvös Loránd Research Network (ELKH), Budapest, Hungary

[2] Department of Manufacturing Science and Engineering, Budapest University of Technology and Economics, Budapest, Hungary

Corresponding author: B. Tipary
Institute for Computer Science and Control (SZTAKI)
H-1111 Budapest, Kende u. 13-17, Hungary
Email: bence.tipary@sztaki.hu

Abstract

Purpose

The purpose of the paper is to give a comprehensive solution method for the manipulation of parts with complex geometries arriving in bulk into a robotic assembly cell. As bin-picking applications are still not reliable in intricate workcells, firstly, the problem is transformed to a semi-structured pick-and-place application, then by collecting and organizing the required process planning steps, a methodology is formed to achieve reliable factory applications even in crowded assembly cell environments.

Design/methodology/approach

The process planning steps are separated to offline precomputation and online planning. The offline phase focuses on preparing the operation, and reducing the online computational burdens. During the online phase, the parts laying in a semi-structured arrangement are first recognized and localized based on their stable equilibrium using 2D vision. Then, the picking sequence and corresponding collision-free robot trajectories are planned and optimized.

Findings

The proposed method was evaluated in a geometrically complex experimental workcell, where it ensured precise, collision-free operation. Moreover, the applied planning processes could significantly reduce the execution time compared to heuristic approaches.

Research limitations/implications

The methodology can be further generalized by considering multiple part types and grasping modes. Additionally, the automation of grasp planning and the enhancement of part localization, sequence planning and path smoothing with more advanced solutions are further research directions.

Originality/value

The paper proposes a novel methodology that combines geometrical computations, image processing and combinatorial optimization, adapted to the requirements of flexible pick-and-place applications. The methodology covers each required planning step to reach reliable and more efficient operation.

Keywords

Robotic manipulator, pick-and-place, object recognition, trajectory planning, optimization

1 Introduction

Despite the growing demand in industry for flexible part handling solutions, the design of flexible robotic pick-and-place operations – where multiple or frequently changing parts, arriving in bulk into the workcell, need to be manipulated – remains a challenge (Holz *et al.*, 2015). Although simple, traditional automation, with predetermined part poses in the workcell, is essentially accomplished, factories address replacing human workforce in more and more complex scenarios. Automating processes with parts in undetermined poses requires advanced robotic pick-and-place solutions that can locate and manipulate the parts based on sensory information. At the same time, flexibility needs to be maintained, even when economic considerations allow only a minimal modification of the original workcell (Ono *et al.*, 2009).

In traditional robot cells, operations assume parts to be laying at a known location in a structured way (e.g., in a pallet), from where they can be grasped and moved to a predefined target pose. When these parts are accessible only in a semi- or unstructured manner (e.g., on trays or in bins in bulk), the first step before further processing is to structure them (Ellekilde and Petersen, 2013). While human workforce can easily cope with indeterminate conditions, these are intractable for traditional approaches in robotics. Also, the application of traditional part-specific feeders (Rosati *et al.*, 2013) is not a viable option in flexible workcells due to the possible part variety and limited room. In these cases, general approaches – most commonly bin-picking (Lozano-Perez *et al.*, 1987) solutions – are recommended. Vision-based systems have shown a great application potential in part recognition (Tsarouchi *et al.*, 2016), inspection (Reinhart and Tekouo, 2009) and robotic handling (Morrison *et al.*, 2018). At the same time, the flexibility of these systems imposes new challenges: namely, the burden of planning the reliable, collision-free, and efficient operation of the workcell using online methodologies. As part manipulation is a non-value added operation, the amount of invested time and effort should be kept minimal.

In order to ensure reliable industrial part manipulation, this paper focuses on the following factory requirements (Kaipa *et al.*, 2016): robust operation with the highest success rate is of key importance, along with minimal computation and execution times. Robot trajectories must be collision-free in every case regarding both the robot-part-environment relations and the robot itself. Furthermore, parts need to be grasped and placed accurately using a general-purpose gripper while allowing complex geometries and multiple different part types. Picking and placing poses, as well as approach directions may be intricate, especially in case of machine feeding. Moreover, workcell designs – often originally optimized for human operations – introduce strict constraints on the robot trajectory.

Consequently, this paper proposes an integrated workflow for planning 2D vision guided robotic pick-and-place operations in a workcell where the loading of the parts is only partially controlled: the part poses are unknown, but it is assumed that a semi-structured arrangement with a suitable separation between the parts can be ensured.

The main contribution of the paper is a comprehensive solution method, divided to offline and online steps, including preparation, part recognition and localization, part sequencing and collision-free trajectory planning. The key features of the proposed approach are (1) visual object localization based on the stable equilibria of the parts, (2) effective optimization of the picking sequence, robot configurations and trajectories, as well as (3) harmonization of the offline and online stages.

2 Related work

2.1 *Flexible pick-and-place applications*

Bin-picking is a highly researched topic of the last decades. The concept being 50 years old (Lozano-Perez *et al.*, 1987), numerous studies were carried out on the fundamental process steps: object detection, recognition and pose estimation, grasp planning and analysis as well as trajectory planning.

Each of these process steps are well studied in the literature. Object detection and pose estimation techniques for grasping are surveyed in (Du *et al.*, 2020). Grasp planning approaches are classified to analytical and data-driven ones; Bicchi and Kumar (2000) give a comprehensive summary on analytical methods, while Bohg *et al.* (2014) on data-driven methods. Recently, attention has shifted towards online grasp planning of novel, everyday objects with universal or multi-functional grippers, for which an overview is given in (Correll *et al.*, 2018).

Regarding complete bin-picking solutions, two main types of scenarios are of interest among researchers. On the one hand, in a warehouse scenario, solutions mainly target the grasping of unknown objects using conventional (Cowley *et al.*, 2013) and machine learning techniques. In this case, the main focus is on improving grasping reliability (Correll *et al.*, 2018; Fujita *et al.*, 2020), with less attention to placement planning. On the other hand, in factory scenarios, the focus is on cycle time reduction together with the tight precision requirements on the picking and placing poses (Harada *et al.*, 2014), while dealing with less uncertainty in the shape of the manipulated objects. However, 3D bin-picking still appears to be a challenging problem, as most successful industrial implementations consider only simple geometric shapes and part seizing using vacuum grippers (Bogue, 2014). The elaboration of more complex scenarios still requires significant engineering work. This indicates the lack of general frameworks and comprehensive methodologies.

In factory scenarios, semi-structured methods seem to increase reliability while allowing higher complexity in part geometries. However, these methods are less studied, and off-the-shelf flexible feeder systems (Waizmann, 2020) offer only partial and costly solution. More intricate scenarios, such as delicate cases where the robot operates in a geometrically dense space or close to singularities, or it must change configurations to cover the full task space, need special attention. These require great effort related to grasp and path planning using precise geometric and kinematic models (Semeniuta *et al.*, 2016).

In the following, relevant papers including factory-related bin-picking applications are overviewed. In the work of Ellekilde and Petersen (2013), an unstructured bin-picking approach is presented. The parts lay in a homogeneous bin with a known location, and during the operation they are placed in a hydraulic press. The focus of this paper is the collision-free path planning and path optimization, which is realized by adapting prepared robot paths to the actual part picking pose. In each cycle a single part (with simple cylindrical geometry) is manipulated, and due to grasping failures, the cycle time occasionally increases. Oh *et al.* (2012) present another unstructured bin-picking approach. In this case, the focus is on the

3D pose estimation of the parts by applying stereo vision and pattern matching, based on the geometric features of the parts. The parts are cylindrical and a vertical approaching motion is employed for part grasping. Although good pose estimation precision is presented, part placement (and corresponding precision) is not mentioned.

In the work of Holz *et al.* (2015), an object detection and localization method is proposed together with path planning for an object depalletizing scenario. Several fixed intermediate poses are used with precomputed trajectories between them to reduce motion times. The manipulation employs simple vertical approach of the parts, and part placement is not considered (the process is stopped at an intermediate pose). In (Su *et al.*, 2016), a piston detection and pose estimation method is presented on a feature recognition basis. This method is applied for piston reorientation and grasping on a conveyor belt, however, the focus is on object pose estimation, whereas collision detection and operation reliability is not mentioned. Part stable equilibrium has been proposed to tackle part identification and localization in (Berger *et al.*, 2000), however, no workflow has been provided regarding the complete pick-and-place operation.

Virtual reality (VR)-based technologies are also suggested for setting up robotic assembly applications. These technologies offer assistance in layout planning, component selection, testing and verification in the design, planning and production phases of a workcell (Pérez *et al.*, 2020). Also, VR is often used for interactive robot programming, and can be beneficial to use for the offline computable parts of the programming, or to verify robot paths (Burghardt *et al.*, 2020). Nevertheless, such systems alone are still not able to solve the core issues of flexible applications, and need to be combined with autonomous online trajectory and sequence planning to achieve reliable workcell operation.

The above papers illustrate that unstructured and semi-structured robotic pick-and-place are still in their immature status where several simplifying assumptions have to be made to ensure reliable operation: typically, part geometries are simple, grasping is performed on a single side of the pieces, no online collision-free path planning is performed, and the required precision is low, especially on the placing side. In addition, the use of intricate metrology systems is commonly required. These issues altogether reduce the applicability of such approaches. The authors are not aware of flexible part manipulation approaches under complex geometrical conditions, and accordingly, of papers in the literature that combine collision-free online path planning, robot configuration changes and task sequencing.

2.2 *Online planning of robotic operations*

Process planning for robotic pick-and-place operations involves the sequencing of the parts, selecting the appropriate grasp for each part, planning the robot trajectories, etc. These decisions are strongly interrelated, and they together determine the performance of the robotic pick-and-place operation. Moreover, they must be made online, with as little computation time as possible.

The optimal sequencing of robotic tasks has received significant attention recently in various application domains, including laser cutting (Dewil *et al.*, 2016), laser welding (Kovács, 2016), and spray painting (Kolakowska *et al.*, 2014). A recent paper (Dewil *et al.*, 2019) emphasizes that in multi-hole drilling, 70% of the total process time is spent in tool movement and tool switching, i.e., in non-effective tasks, thus, optimizing the task sequence and the robot path is crucial for the efficiency of the manufacturing process. The same argument certainly holds in robotic pick-and-place, too. Most of the above contributions model the sequencing problem using an extension of the well-known *Traveling Salesman Problem* (TSP) with the most important domain-specific constraints (Alatartsev *et al.*, 2015). In many robotic applications, there is a certain degree of freedom in how a robotic task is executed; e.g., the positions to visit are fully defined in the task space, but the corresponding robot configurations can be selected freely. Since the selection of the appropriate robot configuration and the task sequence strongly and mutually affect each other, these two sub-problems are often solved together, which is known as the Robotic Task Sequencing Problem (RTSP) (Suárez-Ruiz *et al.*, 2018).

Finally, the collision-free robot path has to be planned in the joint configuration space of the robot. For planning in high dimensional spaces, typically, with six dimensions or more, the application of sampling-based path planners dominates. Hence, path planners in such applications use the single-query Rapidly-exploring Random Tree (RRT) (LaValle and Kuffner, 2001), the multi-query Probabilistic Roadmap (PRM) algorithms (Kavraki *et al.*, 1996), or one of their numerous variants. The performance of various path planning algorithms in bin-picking is investigated in (Iversen and Ellekilde, 2017).

3 Problem setting

This paper addresses the planning of vision-based, semi-structured pick-and-place applications. The 3D geometry of the part, as well as the complete and calibrated geometric and kinematic model of the robotic workcell (Horváth and Erdős, 2017) is considered to be given (see Figure 1(b)). This includes a six degrees of freedom robotic arm with a 2-finger gripper to pick the parts, a 2D camera (with calibrated intrinsic parameters) mounted on the robot, a fixture where the parts must be placed, and a vibratory light table, see Figure 1(a). Parts in bulk are loaded onto the light table to arbitrary picking poses. When several parts overlap or get entangled in poses unreachable for the robot (e.g., in the corner of the light table), then they can be rearranged using the vibratory function of the table.

Then, the planning problem consists of determining the grasping poses of the parts, and computing a picking sequence as well as the corresponding collision-free robot trajectories that minimize the execution time of all pickable parts. A foremost requirement on the solution approach is that this plan can be calculated only online, and hence, with a strict limit on the computation time.

4 Overview of the solution method

In the semi-structured setting, part localization can be performed by matching the 2D camera image to precomputed reference images of the part in its possible stable equilibria. Picking tasks must be planned based on the identified part poses, also taking into account a geometrically accurate workcell model to ensure the reliable and collision-free operation. The complexity of this planning problem could become unmanageable for online calculation; however, if necessary, it is possible to separate a part of the problem that can be solved offline, without knowing the part grasping poses.

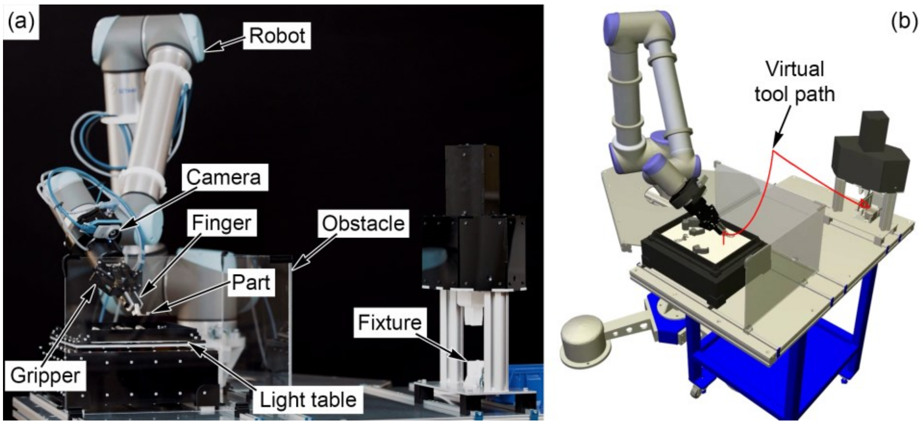

Figure 1. Workcell layout (a) and its virtual kinematic model (b).

For clarification, poses and frames are defined in the robot task space as task space points. However, a pose describes the reference of a rigid body in the task space, whereas frames correspond to a constant transformation relative to the rigid body references (can be multiple per rigid body, such as the part grasp frame). Also, configurations are defined in the robot configuration space as a robot joint vector.

The robot trajectory must start and end in the *image capture pose* (ICP), and pick the parts one-by-one by taking them from their initial, *grasping pose* (GP) to their *placing pose* (PP) corresponding to the fixture. It is assumed that the part is processed and removed from the fixture before the next part is placed there. The sections of the robot trajectory planned online and offline can be separated by introducing an *intermediate pose* (IP) between the GP and the PP. Then, movement from IP to PP (and backwards) becomes precomputable, and online planning is necessary only on the ICP-GP, GP-IP, and IP-next GP trajectories. These steps of the robot operation are depicted in Figure 2, while the poses are shown in Figure 3 (note that no separate IP is defined in this case). *Near-picking* (GP*) and *near-placing poses* (PP*) are noted here, which fulfill the role of handling the contact transient during grasping and placing. Although the near-poses can be different in the approach and retreat part of the manipulation, for the sake of simplicity, these are considered to be identical here. GP* and PP* are located close to the GP and PP, respectively. The planning workflow is presented in Figure 4, while each offline and online computation step is discussed in detail in the coming sections.

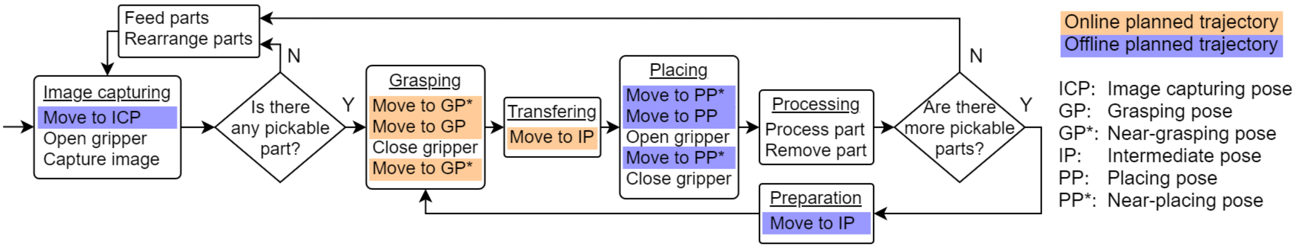

Figure 2. Sequence diagram of the pick-and-place operation.

5 Geometrical precomputation

5.1 Stable equilibrium generation

Stable equilibrium generation for the part departs from a 3D mesh model of the part geometry. While this discrete model is a good approximation of the original geometry, it also allows efficient rigid body calculations. When in a stable equilibrium on a horizontal base plane, a convex polyhedron lays on (at least) one of its stable faces. A face is stable if the projection of the center of gravity (CoG) in the face normal direction pierces through the face area. In case of a concave polyhedron, the stable faces of its convex hull have to be found in order to determine the stable groups of faces, edges or vertices of the original polyhedron (Várkonyi, 2014).

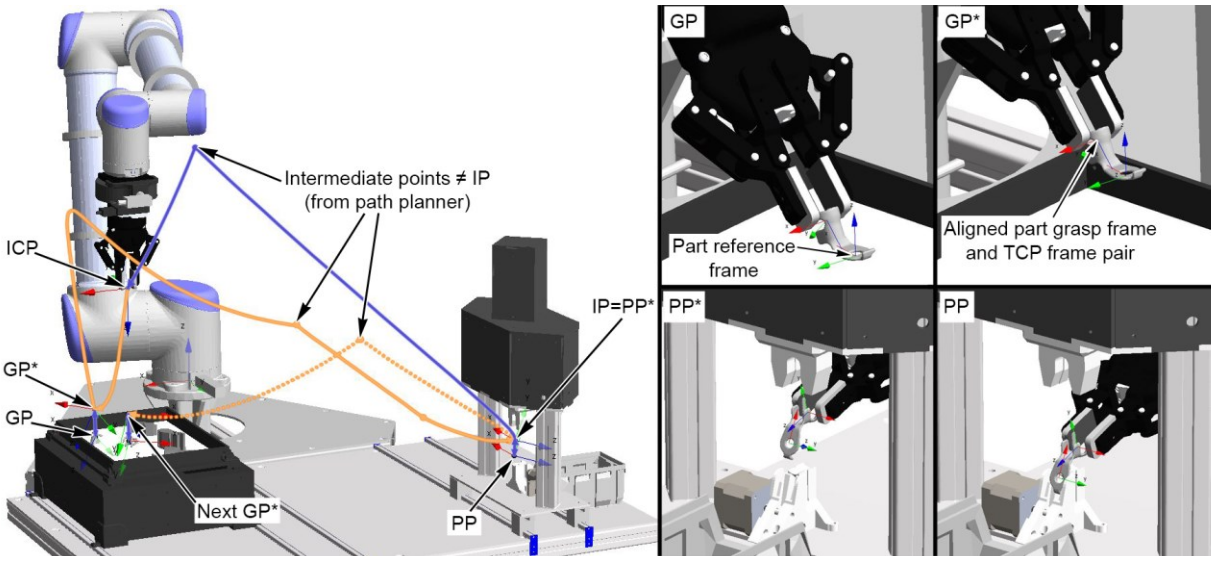

Figure 3. A planned sample tool path, the precomputable segments (dark) and the online planned segments (light) with the corresponding key part poses and robot configurations (note that the obstacle wall is hidden for better visibility).

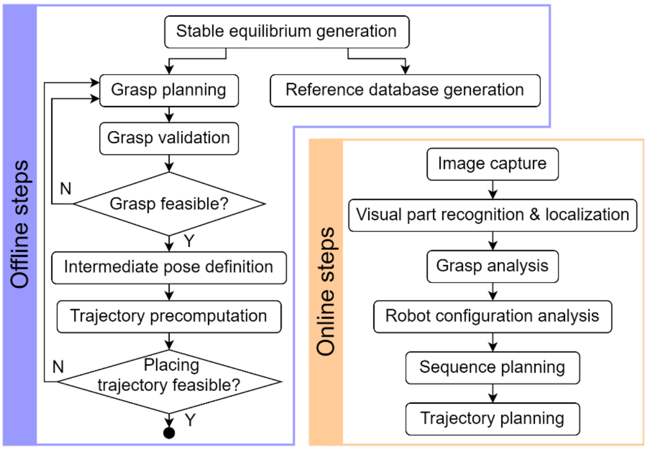
Offline steps
Stable equilibrium generation
Grasp planning
Reference database generation
Grasp validation
Grasp feasible?
N
Y
Intermediate pose definition
Trajectory precomputation
Placing trajectory feasible?
N
Y
Online steps
Image capture
Visual part recognition & localization
Grasp analysis
Robot configuration analysis
Sequence planning
Trajectory planning

Figure 4. Offline and online steps of the workflow.

Consequently, at first, the CoG and the convex hull have to be determined. Next, the stable triangles have to be found on the convex hull. Then, the stable groups on the original part can be determined by face remapping and grouping by connectivity, face normal direction and distance from CoG, considering a certain tolerance. Based on the stable equilibria, the corresponding geometric transformations can be determined. The found equilibria (see Figure 5) are physically validated by applying a small perturbation.

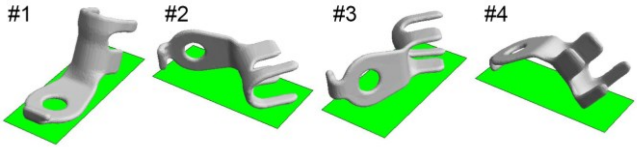

Figure 5. Stable equilibria of a sample cable shoe part.

5.2 *Reference database generation*

In order to enable camera-based stable equilibrium recognition, the physical and virtual environment and imaging system need to be set up (i.e., material properties, light sources, background and image adjustment parameters). The reference database contains perspectively rendered images of the virtual part in the matched workcell environment. These samples are generated with different positions and orientations in a systematic way with a predetermined resolution. The images in the database will be compared to the detected objects in the captured image of the actual parts in the workcell. Additionally, the database contains the part bounding box dimensions, as well as the selected features for localization (image matching and localization are discussed in Section 6.1 in detail).

5.3 Grasp planning

Based on the possible stable equilibria of the part, grasp planning is performed within an iterative process. The goal is to determine the part grasp frame to TCP (tool center point) frame transformation (see Figure 3) alongside with the gripping parameter, i.e., the finger distance, for each *graspable* stable equilibrium, which is selected for grasping. The design process is manual and geometry-based (Spenrath and Pott, 2017), and complies with the basic principles of grasping presented in (Fantoni *et al.*, 2014). This process focuses on avoiding any collisions between the gripper geometry (including the finger, but ignoring the robot) and the obstacles formed by the light table (gripper in GP) and the fixture (gripper in PP). Since GP is not fully known, an infinite plane is considered as a relaxed model of the light table. By combining these environments, the part-gripper relation can be defined (see Figure 6). In addition, GP* and PP* has to be defined for each grasp, which are usually set up as a short linear interpolation in the task space.

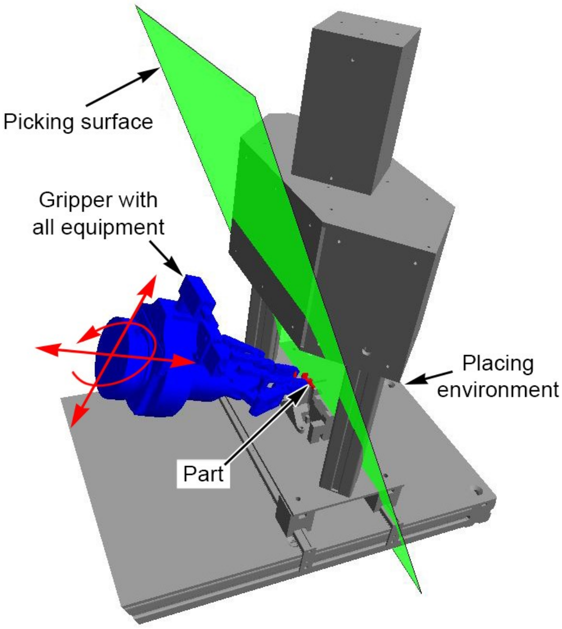

Figure 6. Combination of the picking and placing environments for grasp planning.

5.4 Grasp validation

There are multiple criteria for a planned grasp to be valid. On one hand, the gripper needs to be situated in the workcell without collision in GP and PP (more accurately during the GP-GP* and PP-PP* transitions). On the other hand, the robot needs to reach these poses within its workspace volume without collision, i.e., at least one collision-free inverse kinematic solution (robot configuration) must exist for all the above poses.

Feasibility on the placing side can be ensured with proper workcell design. However, depending on the actual situation of the parts, the gripper may collide with the light table walls or other parts present on the light table, and also a feasible robot configuration might not exist. Therefore, whether or not the above criteria are fulfilled can only be evaluated statistically before the real operation. This can be computed using either random part poses in simulation or based on real data acquired from the real workcell (or part of it), if it is available. If, for a graspable part the above criteria are fulfilled, the part is labeled *pickable*.

If the statistics show that the probability of a part on the light table being pickable is sufficient for the continuous operation, then the grasp plan can be accepted. Otherwise, either the grasp frame-TCP frame relation needs to be revised, or in extreme cases, the workcell design needs to be modified.

5.5 Intermediate pose (IP) definition

IPs can be applied to separate the grasping movement, which must be computed online, from the placing movement, which can be precomputed based on the workcell design. This can be advantageous when the online path planning time is the bottleneck of the system, as this separation facilitates the online path planning process. In this case, IPs are recommended to be defined on the boundary of the geometrically dense area around the fixture (PP) and the free workcell volume. If there are multiple grasp plans (for different graspable stable equilibria), then IPs can be defined for each grasp, separately.

If the environment of the fixture is geometrically sparse, or online path planning is fast, then the specification of IP(s) is not necessary. In fact, the overall robot paths can be shorter without specifying an IP. These cases can be captured with IP= PP*, therefore, both the grasping and the placing motions will be planned online. Consequently, this tradeoff needs to be considered when planning a scenario on a case-by-case basis.

5.6 *Trajectory precomputation*

Robot trajectories must be computed in the robot joint configuration space considering the complete geometric and kinematic workcell model. In order to ensure computational efficiency in the online phase despite the high-dimensional configuration space, a PRM (Kavraki *et al.*, 1996) planning approach is adopted, which computes a map of collision-free configurations and motions between them in the offline precomputation phase.

Moreover, the final trajectory is precomputed for all motions whose endpoints are known in advance. Notably, this includes motions between robot configurations corresponding to IP and the PP* for each graspable stable equilibrium. It is recalled that the transitions between PP* and PP are given in the input associated to fixture design.

The resulting trajectories are smoothed in a post-processing step. The reversed trajectory is applied from PP* back to the same IP, with identical gripper configuration in applications where this is required. If no collision-free trajectory is found, then the given robot configuration is discarded, whereas if all robot configurations are discarded for a graspable stable equilibrium, then grasp planning is re-visited.

6 Online planning and execution

6.1 *Visual part recognition and localization*

The online workflow begins with capturing the camera image of the part arrangement on the light table (see Figure 7(a)). After detecting the present objects, the identification of part stable equilibria and localization is carried out. Exploiting the homogeneous background provided by the light table, the part silhouettes are approximately located and separated from the background by means of thresholding (see Figure 7(b)). Their bounding boxes and the corresponding geometrical center points are also computed. Based on these two characteristics, the reference database is filtered to reduce the number of comparisons.

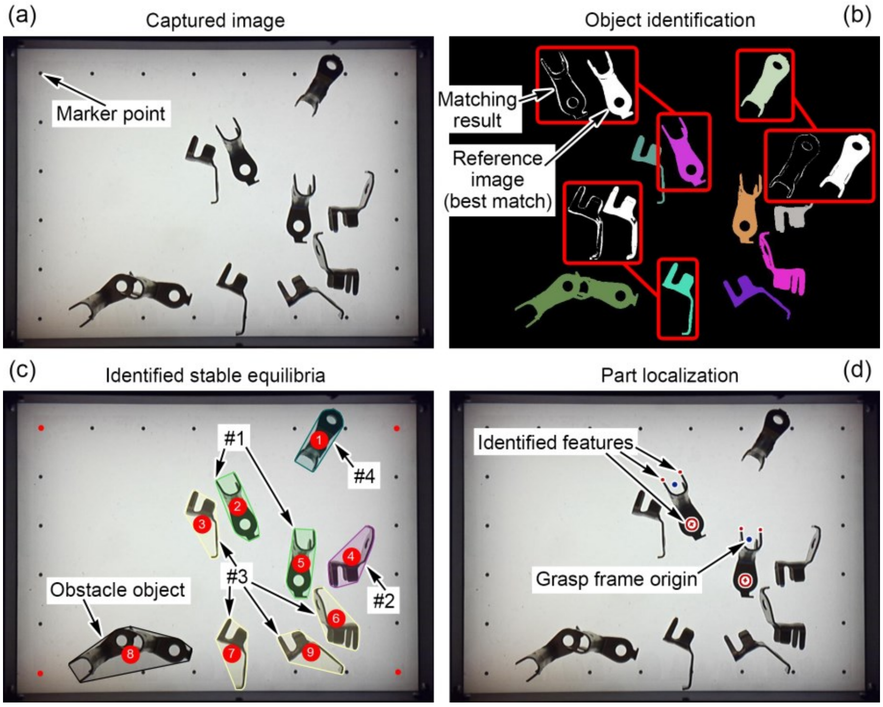

Figure 7. Steps of part identification and localization on a sample image: the captured image (a), object detection and identification (b), the classified objects (c), and feature detection (d).

For the object identification, the filtered virtual images and the segmented camera images are overlaid, and the difference is checked on binary basis (see Figure 7(b)). For each stable equilibrium, a maximum allowed deviation is predetermined. Under this limit, the best-fit reference image is considered to be the recognized image, determining the stable equilibrium. In case there is no reference image within the allowed maximum deviation (e.g., when two parts overlap), the part is considered unidentified, and its perimeter with a predetermined height can be included as an obstacle during grasp analysis.

With this matching method, the separated parts can be identified after applying thresholding and segmentation on the captured image. For not separated parts, parts too close to the light table walls, or other false objects, the identification process classifies the corresponding objects as obstacles. The classified objects are shown in Figure 7(c).

Although the matching process could be used for localization, its accuracy is not sufficient. Therefore, to achieve the desired accuracy in part localization, a feature-based approach is adopted. For this purpose, well distinguishable (groups of) geometrical elements, e.g., holes or edges, are selected on the part for each graspable stable equilibrium, separately.

Part localization is based on the homography transformation (presented by Kazemi *et al.* (2010)) that maps the artificial marker positions (located on the light table) in the camera image to physical 3D positions in the robot coordinate system. Given the calibrated workcell, it is possible to determine the physical position of any further point in the camera image, assuming that the physical height of the particular point is known. Accordingly, the accuracy of the camera mounting position is not critical, i.e., the camera frame does not need to be calibrated relative to the TCP frame.

First, the artificial markers are to be found in the camera image. These provide the reference for the homography transformation. The next step is to find the selected features on the parts that are classified graspable (see Figure 7(d)). With the help of this information, part positions and orientations can be calculated relative to the marker points, and therefore relative to the robot base frame. Using the grasp frames, the corresponding GPs and GP*s can also be calculated. Finally, the virtual workcell model is actualized by inserting the graspable parts in their proper poses along with the prepared obstacle objects (see Figure 8). It is noted that the not graspable parts are also included as obstacles. It is assumed that no external process moves the parts during picking.

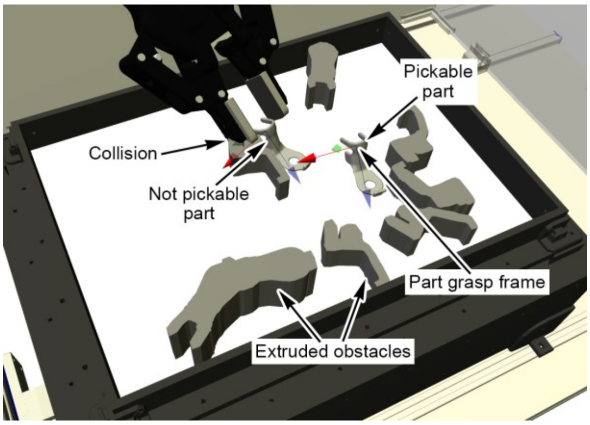

Figure 8. Virtual model update based on a sample image for grasp and robot configuration analysis.

6.2 Grasp analysis

In this step, the first condition of part *pickability* (i.e., whether the part can be feasibly manipulated in the current cycle) is verified for each localized graspable part. For this purpose, the geometric model of the complete gripper is fit to the part in the workcell model (without the robot in this phase), and potential collisions are checked for. If the gripper collides with any static element of the workcell (i.e., any element other than a graspable part; typically, the light table), then the given part is classified as not pickable and handled as a static obstacle in the sequel. Otherwise, if the gripper collides with some other graspable parts, then the corresponding precedence constraints are recorded, which state that the colliding parts must be picked before the actual part.

6.3 Robot configuration analysis

The analysis of the second pickability condition is performed with a focus on the potential collisions of the robot arm along the GP to GP* transition. All robot configurations corresponding to the resolved GPs are generated by solving the inverse kinematic problem, then collision checking is performed. In case of any collision (either with a static element or with another part), the given robot configuration is discarded. If no collision-free configuration remains for a part, then the part itself is again classified as not pickable. Finally, if there are directed cycles of precedences, or parts preceded by a not pickable part, then the affected parts are also discarded.

6.4 *Sequence planning*

Sequence planning addresses computing a minimum-time manipulation sequence for the pickable parts and jointly selecting a robot configuration from the candidates for each pose to visit during picking. The focus is on robot motions GP*-IP-PP* and PP*-IP-next GP*: optimizing this part of the trajectory is important, since potential robot configuration changes impact the manipulation time considerably. In contrast, all other motions (PP*-PP-PP* and GP*-GP-GP*) take generally the same time for each configuration candidate pair; these motions are therefore neglected. Travel times at this phase are estimated by considering linear motions in the joint space using a trapezoid speed profile, and hence, disregarding potential collisions.

This problem is modeled as a *Generalized Traveling Salesman Problem* (GTSP) (Alatartsev *et al.*, 2015; Laporte *et al.*, 1987). In GTSP, nodes are ordered into classes and a tour is sought that visits each class exactly once. Here, classes correspond to poses that must be visited by the robot, whereas nodes within a class stand for the candidate robot configurations for the given pose. Directed edges connect configurations that can follow each other during picking. Hence, there is an edge from the current ICP to the GP* configuration of each part, from each GP* to every IP configuration of the same part, as well as from each IP configuration to every GP* configuration of all other parts and to the next ICP, see Figure 9. Edge costs capture the estimated travel time between the two robot configurations (in case of IP-GP* edges, the time of the complete IP-PP* and PP*-IP-GP* motion sequence). Then, a solution is sought that respects the precedence constraints among the parts and minimizes the cost of the tour. This problem can be solved using professional GTSP or generic Vehicle Routing Problem (VRP) solvers that apply local search with dedicated neighborhoods (Johnson and McGeoch, 1997).

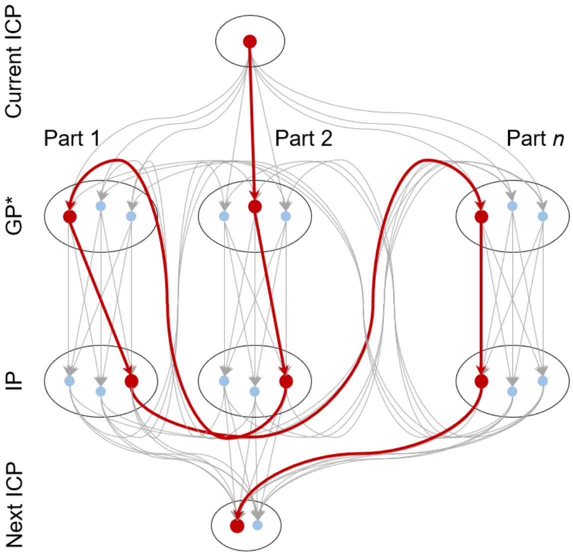

Figure 9. GTSP representation of the sequence planning problem and a solution (bold). Precedence constraints between parts are not displayed.

6.5 *Trajectory planning*

Trajectory planning must be performed online for all motions from the initial ICP to the first GP*, from the GP* to the corresponding IP, as well as from the IP to the next GP*. All other motions are either computed offline or can be determined directly from the identified part locations (GP*-GP-GP* motions). Trajectory planning follows the picking sequence and the selected robot configurations defined during sequence planning and uses the PRM planner and the path smoothing algorithms also applied for offline trajectory precomputation. Trajectories can be computed in parallel with the execution of the previous robot motions and stored in a queue for execution by the robot in the predefined sequence.

7 Implementation and experiments

To evaluate the presented methodology, an experimental robotic workcell was set up. The workcell is motivated by a manual industrial assembly cell for assembling cables and cable shoes with the help of a crimping press. The task is to place the cable shoes into the fixture of the press, then place the cable onto the cable shoe and operate the press to join the two parts. The experimental workcell was developed based on this industrial application, however, with focus on detection, localization and manipulation of the complex geometry of the cable shoes. Manipulation of the cables is ignored. The corresponding experimental workcell with its virtual model is shown in Figure 1. For processing the parts, an imitated press was created, with a programmable ejector to free up the fixture after an imitated pressing process was carried out.

The cable shoes are fed in bulk onto a vibratory light table. The custom-made light table has a switchable LED backlight providing a consistent illumination scene under ordinary day-and-night lighting conditions. The camera settings were left on default, except for the auto focus and the auto gain/shutter features, which were turned off to reduce the image capture time. Using the IDS-XC type 2D camera mounted on the UR10 robot, together with the homogeneous background provided by the light table, the parts are detected, identified and localized. The parts are grasped using the Robotiq 2F-85 finger gripper mounted on the robot. The table vibration and the feed onto the table are configured so that the probability of getting a pickable part is sufficiently high for reliable workcell operation.

The light table is fixed on the worktable well within the robot reach. The press is also fixed on the worktable, however, the press is difficult to access due to its rather closed geometry, as well as due to an additional obstacle wall placed on the worktable (see Figure 1). The workcell is part of a human-robot collaborative workspace, with no fence around the robot. Therefore, robot joint velocities (0.2667 rad/s for each joint) and accelerations (0.6667 rad/s^2 for each joint) are kept low.

The part mesh model was acquired by laser scanning, whereas the virtual workcell model was constructed partly from existing CAD models and partly by reverse engineering. The physical workcell and its virtual counterpart were calibrated. Stable equilibria of the cable shoe were computed, and the corresponding reference database was created. Four stable equilibria were identified that can occur realistically on the vibrational light table, these are designated as #1-#4 according to Figure 5. For three of these equilibria, no grasp could be defined that would be jointly feasible in the GP and the PP without re-grasping. Further experiments were conducted with the single stable equilibrium that had a feasible grasp (#1).

The part grasp frame to TCP frame transformation was selected by combining the picking and placing side environments (see Figure 6). This resulted in a grasp plan where the gripper is tilted 45° from the vertical while picking. In this way, two different collision-free placing configurations could be applied for PP*, and often multiple picking configurations exist for GP*. The ICP also has two collision-free robot configurations. The different candidates are shown in Figure 10.

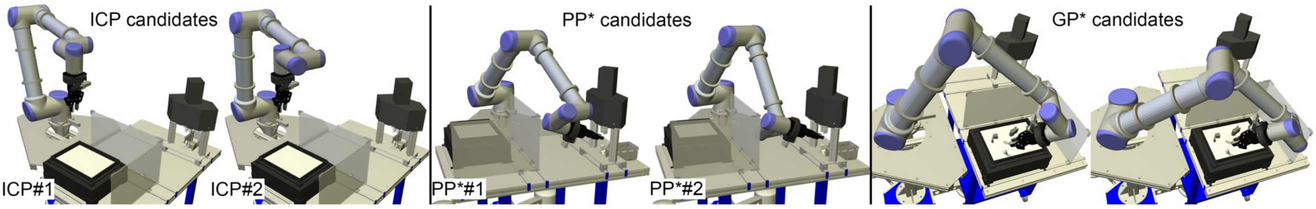

Figure 10. Configuration candidates for ICP, PP* and a sample GP*.

The approach was implemented in the SolidWorks CAD system for 3D modeling, and Wolfram Mathematica for grasp planning, image processing and geometrical computations. LinkageDesigner package was used for kinematical modeling, which allows parametric, motion skeleton-based modeling of kinematic systems with the help of spatial and kinematic constraints. The system architecture of the implementation with the main components, component connections and data flow are shown in Figure 11. The computations were carried out on an Intel i7-4790 CPU @ 3.60GHz PC under Windows 10.

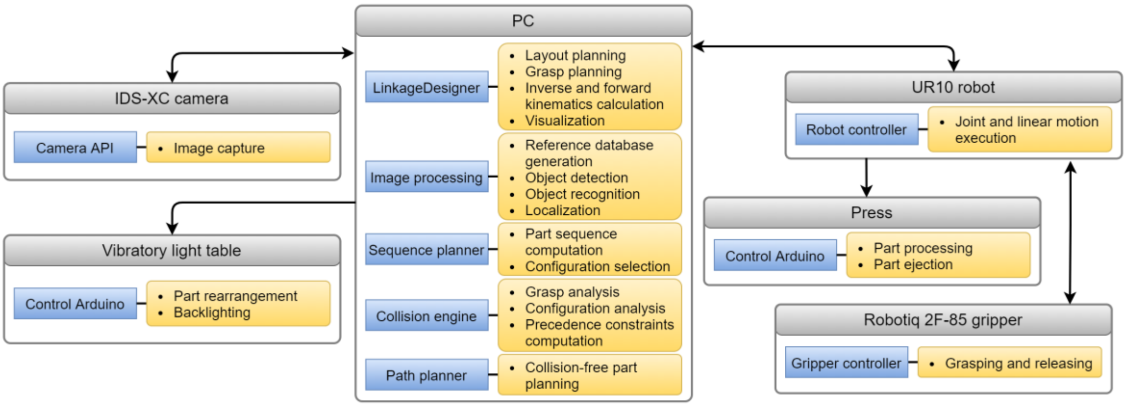

Figure 11. System architecture of the experimental workcell.

The sequence planner was developed on top of the VRP solver of Google OR-Tools, using its default hill climbing search, whereas trajectory planning used the PRM algorithm of the collision detection and path planning library (Zahorán and Kovács, 2019) in C#. A map containing 2000 robot configurations as nodes and 5 neighbors per node was built in the offline precomputation phase. In addition, the PRM map was extended by planning offline paths using virtual parts in sampled positions and orientations. Paths ICP-GP*, GP*-PP* and PP*-GP* were generated and saved in the map for parts laying in positions corresponding to a 4x5 grid on the light table, with 8 different orientations, resulting in altogether 160 cases. During this offline preparation, different types of smoothing were also applied on the paths. As all the corresponding intermediate configurations (calculated by the path planner, not IP, see Figure 3) were saved in the PRM map, the smoothing process during the online path planning could be relaxed, thus, the online planning times could be reduced.

Furthermore, no separate IP specification was necessary for the operation (i.e., IP=PP*), as the path planner could plan the complete paths online. The path planner was implemented so that the next path is always computed during the actual robot motion (except for the first path in the cycle: ICP-GP*). A safety distance of 4 mm was applied. The workcell and the operation is presented in the following video link: https://www.youtube.com/watch?v=9novNg8slN4.

7.1 Physical experiments

Using the above described setup, multiple experiments were carried out. First, the operation was evaluated in the physical space to acquire the online planning and motion times. Altogether 200 parts were manipulated from the vibratory light table into the fixture of the press. This required 217 images, while maintaining 10 parts on the table. After each cycle, the parts were fed back to the light table and the vibratory function of the table was activated to rearrange the parts, and only after a short pause to let the parts settle, was the next cycle begun.

First, the preparation times are presented in Table I. These results show the required precomputation in each case, regardless of having pickable parts or not. Next, computation and robot motion times corresponding to cycles with pickable parts are shown in Table II. It is noted that the path planning time also contains the smoothing time, and as the maximum time suggests, smoothing of intricate paths is the slowest computational process. In general, little smoothing is required for the first path planning, and subsequent smoothing processes can be performed in parallel with the actual robot motion.

Table I. Measured computation times for the online preparation processes for all 217 cycles.

Process	Mean time [s]	Std. time [s]	Min. time [s]	Max. time [s]
Image capture	0.621	0.007	0.597	0.640
Marker detection	0.628	0.049	0.462	0.781
Object detection	0.889	0.050	0.756	1.153
Object recognition	0.351	0.097	0.153	0.773
Localization	0.904	0.445	0.060	2.367
Total preparation	3.531	0.445	2.426	4.742

In addition, vibration and waiting take 1.5 s per image.

Table II. Measured computation times required until the first robot motion, as well as the measured motion times for the 121 cycles with pickable parts.

Process	Mean time [s]	Std. time [s]	Min. time [s]	Max. time [s]
Grasp analysis	0.245	0.099	0.073	0.548
Configuration analysis	0.60	0.304	0.303	1.540
Sequence planning	0.076	0.004	0.07	0.091
Path planning (until 1st motion)	0.315	0.894	0.055	4.678
Total until 1st motion (including preparation)	4.759	1.218	2.976	10.280
First picking motion	9.840	4.580	4.327	22.694
Placing motion	14.079	4.397	6.836	24.182
Next picking motion	12.769	2.626	8.564	24.182

Using trapezoid velocity profiles, real and simulated motion times matched with 99.7% precision. Robot paths from PP* to ICP are offline computed and the PP*#1-ICP#1 motion (see Figure 10) takes 12.377±0.010 s, and the PP*#2-ICP#2 motion takes 15.129±0.010 s. During the 121 cycles, the ICP#1 configuration was selected by the sequence planner in 97 cases, and the ICP#2 in 24 cases.

Finally, the total robot motion times per manipulated parts are presented for different number of pickable parts in a single cycle in Table III. The results imply that the more parts are manipulated in a single cycle the lower the average manipulation time per part is. This is because the time required for the PP*-ICP motion at the end of the cycle, and for ICP-GP* in the beginning of the next cycle are higher than that of a single PP*-GP* motion. When considering also the average preparation and robot motion time per part manipulation, it becomes evident that the more parts are manipulated in a single cycle, the more efficient the system becomes.

Although during the measurements different lighting conditions were present (altering daylight and room lighting), the part identification and localization processes were robust and did not produce any false GPs. Furthermore, no failures (collisions, part drops, etc.) occurred during the operation, and the part placement was successful in each manipulation. In some cases, the cables running along the robot arm rubbed against the obstacle wall or the robot joints, but no tangling or stretching occurred. Overall, these results show and prove the reliability of the application.

Table III. Measured motion times per number of pickable parts in a single cycle.

No. of pickable parts [pc.]	Occurrence no. [pc.]	Mean time [s]	Std. time [s]	Min. time [s]	Max. time [s]
1	66	36.496	5.295	27.278	54.374
2	33	32.235	3.338	27.643	40.840
3	20	29.744	3.987	22.526	36.521
4	2	29.910	1.431	28.898	30.923
For all cases	121	34.109	5.316	22.526	54.374

7.2 Simulation-based experiments

After the physical experiments, a set of 500 images were captured of 10 parts placed on the vibratory light table. This was carried out in the same way as during the experiments in the physical cell, but without performing the other online steps and robot motions. Although this experiment could have been prepared using sampled part positions and orientations in the virtual model, this solution was preferred, as the workcell was already established, and the characteristics of the part rearrangement and blocking of part grasping by other parts are better represented in this way.

With the help of this real dataset, the pick-and-place process was simulated, including every step of the online planning process. Firstly, part pickability was analyzed, for which the results are presented in Table IV. The results show that 34.06% of the parts were in graspable stable equilibrium. Although this value could be improved by tuning the vibration strategy, it was accepted for operation.

Table IV. Simulation results on the number of pickable parts for 500 captured photos, each containing 10 parts.

	Correctly processed parts [pc.]	**Failed parts [pc.]**
Total number of parts	**5000**	
No visible markers on image (2 images)		20
Unidentified objects: tangled or on wall		1457
Unidentified objects: parts not settled		5
Unidentified objects: similar looking objects		84
Identified	**3434**	
Not graspable		1731
Graspable	**1703**	
Localization failed		19
Localized	**1684**	
Not pickable: gripper collision with environment		511
Not pickable: gripper collision with other parts		625
Not pickable: no collision-free configuration		38
Pickable	**510**	

There were 84 cases where the object identification failed because of the similarity of two stable equilibria from particular viewpoints (#1 and #4 in Figure 5). Stable equilibria that matched closely with the same object were necessary to filter and consider the object as an obstacle in order to avoid producing false GPs, which would reduce system reliability. In further 19 cases, the localizing features could not be found on the identified objects. This phenomenon occurred when the silhouette of these features overlapped with one of the marker points or with the light table walls, or when wrong features were found, resulting in evidently incorrect part positions or orientations, which were automatically filtered out.

From the correctly localized, graspable parts, 30.29% were pickable. The pickability was mainly reduced by the light table walls and other parts, preventing collision-free grasping (one such case is shown in Figure 8). Also, in 38 cases, parts were not pickable due to the lack of collision-free configuration for GP.

In the following, using the same dataset, the effects of sequence planning on the robot motion times are analyzed. For this, the simulation of the online process steps was repeated without optimized sequence planning. In this case, all precedences are considered similarly as in case of sequence planning, however, the part sequence and corresponding GP*, PP* and final ICP configurations are selected based on the closest next configuration (where the closest configuration is the one with the smallest maximum joint deviation in the joint space, neglecting possible collisions). This approach is referred to as *closest first heuristic*. The results of the simulations are shown in Table V as ratios of the robot motion times without and with the optimized sequence planning approach. Hence, a ratio below one indicates that effective sequence planning could decrease execution time compared to the heuristic approach. As there are two ICP candidates, the results were simulated using the results of the physical experiments, i.e., the cycles were simulated with a 97:24 ratio. From the 315 photos that contained pickable parts, 253 were evaluated with ICP#1, and 62 with ICP#2 being the initial configuration.

Table V. Simulation results on the benefits of applying sequence planner (time ratio = motion time with optimized sequence planning divided by motion time with closest first heuristic).

No. of pickable parts [pc.]	Occurrence no. [pc.]	Mean time ratio [-]	Std. time ratio [-]	Min. time ratio [-]	Max. time ratio [-]
0	185	-	-	-	-
1	172	0.9631	0.1152	0.6509	1.2986
2	101	0.9778	0.1276	0.7249	1.1794
3	34	0.9173	0.0979	0.6580	1.0794
4	7	0.9708	0.0755	0.8762	1.0724
5	0	-	-	-	-
6	1	0.8969	-	-	-
For all cases	500	0.9628	0.1176	0.6509	1.2986

As can be seen in Table V, 3.72% mean improvement was realized using the sequence planner compared to the heuristic approach. The results were separated for the number of pickable parts in a single cycle. Improvement can also occur in case of one pickable part, because of the multiple selectable configuration candidates. The time reduction seems to be greater with the higher amount of pickable parts, however, this phenomenon needs to be analyzed further in the future due to the low occurrence number. In some cases, the time ratio was larger than one. This can arise because the sequence planner works with estimated motion times, before computing the collision-free paths between the pairs of configurations.

Finally, using the simulation dataset, reasonable alternative approaches were compared in terms of average manipulation and execution time per parts. Manipulation times only include the robot motion, while execution times also include the part rearrangement and the total preparation times. The scenarios were defined with the following specifications: which key path points are allowed to have multiple configuration candidates, what IP is applied, which paths are planned offline and online and how is the sequence planned. The different strategies and corresponding results are shown in Table VI. Each of the approaches assume that reliable, virtual model-based collision detection is available, otherwise the operation could not be guaranteed due to the intricate robot configurations and constrained environment.

Table VI. Simulation results for different manipulation strategies.

Strategy	Multiple selectable config.	IP	Offline paths	Online paths	Sequence planning	No. of pickable parts [pc.]	Overall mean manipulation time per parts [s]	Overall mean execution time per parts [s]
I.	GP*	IP=ICP	ICP-PP*, PP*-ICP	ICP-GP^{*1}, GP*-ICP^{1}	Closest first	470	40.246	45.256
II.	GP*	IP=IP1	ICP-IP, IP-PP*, PP*-ICP	IP-GP^{*1}, GP*-IP^{1}	Closest first	295	42.359	50.496
III.	GP*	IP=IP2	ICP-IP, IP-PP*, PP*-ICP	IP-GP^{*1}, GP*-IP^{1}	Closest first	470	39.474	44.484
IV.	GP*	IP=IP2	ICP-IP, IP-PP*, PP*-ICP	ICP-GP^{*2} (for first part), IP-GP^{*2}, GP*-IP^{2}	Closest first	510	40.662	45.519
V.	ICP, GP*, PP*	IP=PP*	GP*-ICP	ICP-GP^{*2}, GP*-PP^{*2}	Closest first	510	35.154	40.010
VI.	ICP, GP*, PP*	IP=PP*	GP*-ICP	ICP-GP^{*2}, GP*-PP^{*2}	Optimized	510	33.564	38.496

IP1: TCP in the middle of the light table, 130 mm above the light table (below the top of the obstacle wall)
IP2: TCP in the middle of the light table, 220 mm above the light table (above the top of the obstacle wall)
[1] joint interpolation with collision check
[2] collision-free path planning

In strategies I-III., no real online path planning was applied, only joint interpolations were considered with collision prediction along the path. In case of a collision, the involved part is discarded from the pickable parts. This can be considered as an alternative, relaxed path planning strategy. The difference between these strategies is the selected IP, which is specified at different heights above the light table. In strategy IV., an IP was used together with collision-free online path planning. Finally, strategies V. and VI. were essentially the same as discussed at the sequence planning experiment. Here are multiple configuration candidates first introduced for ICP and PP*, without using a specific IP. Strategy V. uses the closest first heuristic approach for sequence planning, while VI. uses the optimized sequence planning approach.

The results of I-III. show that the specification of the IP is not a trivial task. Using an IP generally closer to the GP* candidates does not seem to reduce the motion time significantly. In fact, when placing IP too close to GP*, the cycle times start increasing. The reason for this is that by shortening the IP-GP* distance, the joint distances are not necessarily reduced to the same extent, however, the motion time of the PP*-IP paths increase. Furthermore, when the IP is too close to GP*, the number of pickable parts (without proper online path planning) reduces. Therefore, since less parts are manipulated in a single cycle, efficiency deteriorates further.

When comparing strategy III. to IV., with the difference being the application of online path planning, it is visible that the number of pickable parts increases, however, so does the average part manipulation time, since some of the new pickable parts result in longer paths. Moreover, the path planning time is also added to the preparation time, which results in a somewhat worse manipulation efficiency than III. Nevertheless, when online collision-free path planning is applied, the possibilities for multiple configuration candidates open up (with only joint interpolation and collision checking, switching between configurations is rarely possible). By introducing multiple candidates for ICP and PP*, investing computation time for path planning pays off, as the number of candidates for the closest next configuration increases, and the heuristic sequence planning approach provides shorter overall motion paths. Finally, sequence planning reduces the cycle times even further, as discussed previously.

These results show that the proposed combination of approaches is beneficial, as the average execution time for part manipulation can be reduced significantly. The definition of IP needs to be considered on a case-by-case basis. For example, if the robot joint velocities and accelerations were increased to such an extent that online path planning could not be realized during robot motion, the application of IP would become a viable option, especially using online path planning and multiple configuration candidates. Also, the longer the preparation time (including part rearrangement) of each cycle, the more beneficial it is to pick multiple parts in a single cycle. That is, the application of online path planning and sequence planning is more and more advantageous.

8 Conclusion and future work

This paper proposed a comprehensive methodology for the planning of robotic pick-and-place operations in a semi-structured environment based on 2D vision. The approach combined offline precomputation and online planning using geometrical computations, image processing, and combinatorial optimization. The emphasis was on ensuring reliable and collision-free operation to meet factory requirements even for complex parts and highly constrained workcells requiring special care. Effective optimization techniques for planning the picking sequence and the robot path minimize the execution times. The full-fledged digital simulation model enables a meticulous assessment of the robotic pick-and-place cell already in the design stage, before its actual physical implementation, from the viewpoints of part pickability and execution times. This way, the model can also support cell layout design and verification.

The approach was evaluated in an experimental implementation based on an industrial use case. Using this methodology, reliable and precise operation was achieved in the experimental workcell. It was shown that collision-free path planning together with multiple configuration candidates along the robot trajectory and optimized sequence planning can significantly reduce the execution time compared to heuristic approaches.

Future work will focus on enhanced support for grasp planning; multiple candidate grasps for each equilibrium of the part; efficient part localization using deep learning techniques; stronger integration of sequence and path planning; and the development of an adaptive path smoothing algorithm. The generalization of the model in different directions, such as to multiple part types or to different assembly technologies is a further possible research direction.

Acknowledgements

The authors thank László Zahorán, Máté Bodor and Balázs Németh for their contributions to the implementation of the collision detection, path planning, and sequencing algorithms used in the paper. Work in this paper has been in part funded under project number ED_18-22018-0006, supported by the National Research, Development and Innovation Fund of Hungary, financed under the (publicly funded) funding scheme according to Section 13. §(2) of the Scientific Research, Development and Innovation Act, and in part by the Ministry for Innovation and Technology and the National Research, Development and Innovation Office within the framework of the National Lab for Autonomous Systems. A. Kovács acknowledges the support of the János Bolyai scholarship.

References

Alatartsev, S., Stellmacher, S. and Ortmeier, F. (2015), "Robotic task sequencing problem: a survey", *Journal of Intelligent & Robotic Systems*, Vol. 80 No. 2, pp. 279–298. https://doi.org/10.1007/s10846-015-0190-6

Berger, M., Bachler, G. and Scherer, S. (2000), "Vision guided bin picking and mounting in a flexible assembly cell", in Logananthara, R., Palm, G. and Ali, M. (Ed.s), *Intelligent Problem Solving. Methodologies and Approaches IEA/AIE 2000*, Vol. 1821, Springer, Berlin, Heidelberg, pp. 109–117. https://doi.org/10.1007/3-540-45049-1_14

Bicchi, A. and Kumar, V. (2000), "Robotic grasping and contact: a review", *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, Vol. 1, IEEE, San Francisco, CA, USA, pp. 348–353. https://doi.org/10.1109/ROBOT.2000.844081

Bogue, R. (2014), "Random bin picking: has its time finally come?", *Assembly Automation*, Vol. 34 No. 3, pp. 217–221. https://doi.org/10.1108/AA-03-2014-024

Bohg, J., Morales, A., Asfour, T. and Kragic, D. (2014), "Data-driven grasp synthesis – a survey", *IEEE Transactions on Robotics*, Vol. 30 No. 2, pp. 289–309. https://doi.org/10.1109/TRO.2013.2289018

Burghardt, A., Szybicki, D., Gierlak, P., Kurc, K., Pietruś, P. and Cygan, R. (2020), "Programming of industrial robots using virtual reality and digital twins", *Applied Sciences*, Vol. 10 No. 2, p. 486. https://doi.org/10.3390/app10020486

Correll, N., Bekris, K.E., Berenson, D., Brock, O., Causo, A., Hauser, K., Okada, K., *et al.* (2018), "Analysis and observations from the first Amazon Picking Challenge", *IEEE Transactions on Automation Science and Engineering*, Vol. 15 No. 1, pp. 172–188. https://doi.org/10.1109/TASE.2016.2600527

Cowley, A., Cohen, B., Marshall, W., Taylor, C.J. and Likhachev, M. (2013), "Perception and motion planning for pick-and-place of dynamic objects", *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Tokyo, pp. 816–823. https://doi.org/10.1109/IROS.2013.6696445

Dewil, R., Küçükoğlu, İ., Luteyn, C. and Cattrysse, D. (2019), "A critical review of multi-hole drilling path optimization", *Archives of Computational Methods in Engineering*, Vol. 26 No. 2, pp. 449–459. https://doi.org/10.1007/s11831-018-9251-x

Dewil, R., Vansteenwegen, P. and Cattrysse, D. (2016), "A review of cutting path algorithms for laser cutters", *The International Journal of Advanced Manufacturing Technology*, Vol. 87 No. 5, pp. 1865–1884. https://doi.org/10.1007/s00170-016-8609-1

Du, G., Wang, K., Lian, S. and Zhao, K. (2020), "Vision-based robotic grasp detection from object localization, object pose estimation to grasp estimation: a review", *ArXiv:1905.06658*, pp. 1–24.

Ellekilde, L.-P. and Petersen, H.G. (2013), "Motion planning efficient trajectories for industrial bin-picking", *The International Journal of Robotics Research*, Vol. 32 No. 9–10, pp. 991–1004. https://doi.org/10.1177/0278364913487237

Fantoni, G., Santochi, M., Dini, G., Tracht, K., Scholz-Reiter, B., Fleischer, J., Kristoffer Lien, T., *et al.* (2014), "Grasping devices and methods in automated production processes", *CIRP Annals*, Vol. 63 No. 2, pp. 679–701. https://doi.org/10.1016/j.cirp.2014.05.006

Fujita, M., Domae, Y., Noda, A., Garcia Ricardez, G.A., Nagatani, T., Zeng, A., Song, S., *et al.* (2020), "What are the important technologies for bin picking? Technology analysis of robots in competitions based on a set of performance metrics", *Advanced Robotics*, Vol. 34 No. 7–8, pp. 560–574. https://doi.org/10.1080/01691864.2019.1698463

Harada, K., Tsuji, T., Nagata, K., Yamanobe, N. and Onda, H. (2014), "Validating an object placement planner for robotic pick-and-place tasks", *Robotics and Autonomous Systems*, Vol. 62 No. 10, pp. 1463–1477. https://doi.org/10.1016/j.robot.2014.05.014

Holz, D., Topalidou-Kyniazopoulou, A., Stuckler, J. and Behnke, S. (2015), "Real-time object detection, localization and verification for fast robotic depalletizing", *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Hamburg, Germany, pp. 1459–1466. https://doi.org/10.1109/IROS.2015.7353560

Horváth, G. and Erdős, G. (2017), "Point cloud based robot cell calibration", *CIRP Annals*, Vol. 66 No. 1, pp. 145–148. https://doi.org/10.1016/j.cirp.2017.04.044

Iversen, T.F. and Ellekilde, L.-P. (2017), "Benchmarking motion planning algorithms for bin-picking applications", *Industrial Robot: An International Journal*, Vol. 44 No. 2, pp. 189–197. https://doi.org/10.1108/IR-06-2016-0166

Johnson, D.S. and McGeoch, L.A. (1997), "The traveling salesman problem: a case study in local optimization", *Local Search in Combinatorial Optimization*, Chichester, UK, Vol. 1 No. 1, pp. 215–310.

Kaipa, K.N., Kankanhalli-Nagendra, A.S., Kumbla, N.B., Shriyam, S., Thevendria-Karthic, S.S., Marvel, J.A. and Gupta, S.K. (2016), "Addressing perception uncertainty induced failure modes in robotic bin-picking", *Robotics and Computer-Integrated Manufacturing*, Vol. 42, pp. 17–38. https://doi.org/10.1016/j.rcim.2016.05.002

Kavraki, L.E., Svestka, P., Latombe, J.-C. and Overmars, M.H. (1996), "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE Transactions on Robotics and Automation*, Vol. 12 No. 4, pp. 566–580. https://doi.org/10.1109/70.508439

Kazemi, M., Gupta, K. and Mehrandezh, M. (2010), "Path-planning for visual servoing: a review and issues", in Chesi, G. and Hashimoto, K. (Ed.s), *Visual Servoing via Advanced Numerical Methods*, Vol. 401, Springer, London, pp. 189–207. https://doi.org/10.1007/978-1-84996-089-2_11

Kolakowska, E., Smith, S.F. and Kristiansen, M. (2014), "Constraint optimization model of a scheduling problem for a robotic arm in automatic systems", *Robotics and Autonomous Systems*, Vol. 62 No. 2, pp. 267–280. https://doi.org/10.1016/j.robot.2013.09.005

Kovács, A. (2016), "Integrated task sequencing and path planning for robotic remote laser welding", *International Journal of Production Research*, Vol. 54 No. 4, pp. 1210–1224. https://doi.org/10.1080/00207543.2015.1057626

Laporte, G., Mercure, H. and Nobert, Y. (1987), "Generalized travelling salesman problem through n sets of nodes: the asymmetrical case", *Discrete Applied Mathematics*, Vol. 18 No. 2, pp. 185–197. https://doi.org/10.1016/0166-218X(87)90020-5

LaValle, S.M. and Kuffner, J.J. (2001), "Rapidly-exploring random trees: progress and prospects", *Algorithmic and Computational Robotics: New Directions*, Vol. 5, pp. 293–308.

Lozano-Perez, T., Jones, J., Mazer, E., O'Donnell, P., Grimson, W., Tournassoud, P. and Lanusse, A. (1987), "Handey: a robot system that recognizes, plans, and manipulates", *1987 IEEE International Conference on Robotics and Automation (ICRA)*, Vol. 4, pp. 843–849. https://doi.org/10.1109/ROBOT.1987.1087847

Morrison, D., Leitner, J. and Corke, P. (2018), "Closing the loop for robotic grasping: a real-time, generative grasp synthesis approach", *Robotics: Science and Systems XIV*, Robotics: Science and Systems Foundation https://doi.org/10.15607/RSS.2018.XIV.021

Oh, J.-K., Lee, S. and Lee, C.-H. (2012), "Stereo vision based automation for a bin-picking solution", *International Journal of Control, Automation and Systems*, Vol. 10 No. 2, pp. 362–373. https://doi.org/10.1007/s12555-012-0216-9

Ono, K., Hayashi, T., Fujii, M., Shibasaki, N. and Sonehara, M. (2009), "Development for industrial robotics applications", *IHI Engineering review*, Vol. 42 No. 2, pp. 103–107.

Pérez, L., Rodríguez-Jiménez, S., Rodríguez, N., Usamentiaga, R. and García, D.F. (2020), "Digital twin and virtual reality based methodology for multi-robot manufacturing cell commissioning", *Applied Sciences*, Vol. 10 No. 10, p. 3633. https://doi.org/10.3390/app10103633

Reinhart, G. and Tekouo, W. (2009), "Automatic programming of robot-mounted 3D optical scanning devices to easily measure parts in high-variant assembly", *CIRP Annals*, Vol. 58 No. 1, pp. 25–28. https://doi.org/10.1016/j.cirp.2009.03.125

Rosati, G., Faccio, M., Carli, A. and Rossi, A. (2013), "Fully flexible assembly systems (F-FAS): a new concept in flexible automation", *Assembly Automation*, Vol. 33 No. 1, pp. 8–21. https://doi.org/10.1108/01445151311294603

Semeniuta, O., Dransfeld, S. and Falkman, P. (2016), "Vision-based robotic system for picking and inspection of small automotive components", *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 549–554. https://doi.org/10.1109/COASE.2016.7743452

Spenrath, F. and Pott, A. (2017), "Gripping point determination for bin picking using heuristic search", *Procedia CIRP*, Vol. 62, pp. 606–611. https://doi.org/10.1016/j.procir.2016.06.015

Su, J., Liu, Z.-Y., Qiao, H. and Liu, C. (2016), "Pose-estimation and reorientation of pistons for robotic bin-picking", *Industrial Robot: An International Journal*, Vol. 43 No. 1, pp. 22–32. https://doi.org/10.1108/IR-06-2015-0129

Suárez-Ruiz, F., Lembono, T.S. and Pham, Q.-C. (2018), "RoboTSP – a fast solution to the robotic task sequencing problem", *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Brisbane, QLD, pp. 1611–1616. https://doi.org/10.1109/ICRA.2018.8460581

Tsarouchi, P., Matthaiakis, S.-A., Michalos, G., Makris, S. and Chryssolouris, G. (2016), "A method for detection of randomly placed objects for robotic handling", *CIRP Journal of Manufacturing Science and Technology*, Vol. 14, pp. 20–27. https://doi.org/10.1016/j.cirpj.2016.04.005

Várkonyi, P.L. (2014), “Estimating part pose statistics with application to industrial parts feeding and shape design: new metrics, algorithms, simulation experiments and datasets”, *IEEE Transactions on Automation Science and Engineering*, Vol. 11 No. 3, pp. 658–667. https://doi.org/10.1109/TASE.2014.2318831

Waizmann, S. (2020), “See, recognize, grasp”, *PhotonicsViews*, Vol. 17 No. 4, pp. 52–54. https://doi.org/10.1002/phvs.202070413

Zahorán, L. and Kovács, A. (2019), “Efficient collision detection for path planning for industrial robots”, in Fister Jr., I., Brodnik, A. and Krnc, M. (Ed.s), *StuCoSReC. Proceedings of the 2019 6th Student Computer Science Research Conference*, University of Primorska Press, pp. 19–22. https://doi.org/10.26493/978-961-7055-82-5.19-22