Robotic Tight Packaging using a Hybrid gripper with Variable Stiffness*

Michele Moroni[1,2,+], Ana Elvira Huezo Martin[1,+][0000-0003-4127-7762], Leonard Klüpfel[1][0009-0002-9056-5912], Ashok M. Sundaram[1][0000-0001-9201-6947], Werner Friedl[1][0000-0003-3002-7274], Francesco Braghin[2][0000-0002-0476-4118], and Máximo A. Roa[1][0000-0003-1708-4223]

[1] German Aerospace Center (DLR), Münchner Str. 20, 82234 Weßling, Germany.
{name.surname}@dlr.de
[2] Politecnico di Milano, Via La Masa, 1, 20156 Milano, Italy.
michele2.moroni@mail.polimi.it, francesco.braghin@polimi.it

Abstract. The involvement of robots in warehouse automation poses new problems to research in logistic tasks such as tight packaging, in which a container must be completely filled with items, in a regular and ordered manner, leaving minimum clearance between them. This work investigates the effect of a reliable placing strategy using a system with passive compliance to improve robustness and success rate in such a task. The methodology is integrated into a full pipeline to execute the packaging operation and is evaluated in a real robot, using a mechanically compliant hybrid gripper with variable stiffness, exploring the roles of the hand configuration and stiffness level in the task execution. Along different evaluation tasks, the results show an improvement in success rate thanks to a reliable insertion strategy, when compared to a trivial one. They also demonstrate the efficacy of using variable stiffness to reduce error propagation.

Keywords: Robotic Packaging · Variable Stiffness · Hybrid Gripper.

1 Introduction

The introduction of robotics in logistic scenarios poses new research challenges, mainly in the development of end-effectors, and methodologies for item manipulation and packaging. For packaging applications, the focus is mainly on checking if an item fits in the bin and finding a suitable placement for it. This aspect is often solved as trivially as lowering vertically the object into its intended position and releasing it [13], [19]. Most approaches deal with uncertainty by leaving some margin between objects, which results in loose and irregular packages [2]. A more challenging and realistic application happens when the robot performs tight packaging, meaning that the container is filled with items so that they fit closely together, with minimum clearance between them (Fig. 1). This setting however provides a contact-rich environment, with noise and disturbances coming from either the object localization, the perception of the bin, or the robot and end-effector themselves.

* +Equal contribution to this work. *This work was partially funded by the European Commission under the Horizon Europe Framework Programme grant number 101070600, project SoftEnable.

Fig. 1: Robotic tight packaging task and our experimental setup.

The tight packaging task can be subdivided into three smaller operations: object grasping, package planning and item insertion.

Object grasping. For the robot to get a hold on the object, suction cups have proved a valuable tool, both for grasping [3] and manipulating [4] objects. However, using suction is not always possible, hence we propose to use hybrid fingers to broaden the range of objects the gripper can work with. In our previous work [5], we proposed a planner to select the best grasp modality using hybrid grippers.

Package Planning. Also known as Bin Packing, this problem has been considered since the 80s using different heuristics [1]. More recently, the research has focused toward online packing planning. [17] proposes a methodology to verify if a given item set can fit in the container, regardless of the order of the objects. [7] implements a pipeline to evaluate online the dimension of unknown items arriving with unknown order, and to find a placement for them using a distance-based heuristic; [18] solves the same task with a height-based heuristic.

Item insertion. Tight packaging is similar to a less constrained Peg-in-Hole problem. Late solutions for peg-in-hole use visual feedback [12], force feedback [16], and some estimation of the contact state [11]. These works highlight the importance of exploiting contacts with the environment and using intrinsic compliance to guide the insertion, an idea that we follow in this paper.
A full pipeline was proposed in [15] to solve the tight packaging problem using a suction cup and visual feedback; to place an item, a simple insertion is performed in a free portion of the container, and then the object is moved toward the desired position. Corrective actions are also implemented to fix imprecise placements, pushing and pulling the item. In this way, a tight package is achieved, but it is not possible to fill the whole container, as it is necessary to have space to insert the item and maneuver it. [20] uses instead a clawed gripper to place cubes using force/torque sensing to guide the execution. The approach improves on the previous one by increasing the placement accuracy (reducing the need for corrective actions). However, the chosen gripper also prevents the complete filling of the container.

Our work deals with the tight packaging problem by 1) proposing a package planner integrated with a bio-inspired insertion strategy that only requires the

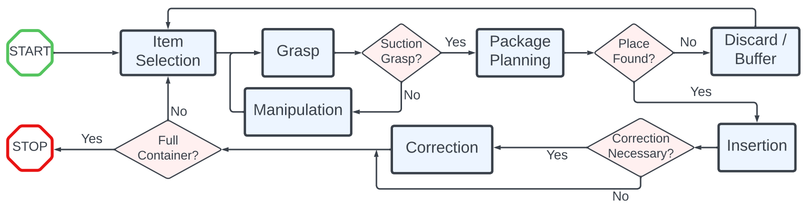

Fig. 2: Scheme of the execution of the task.

intended place for the item to be free, hence filling the container up to all four walls; 2) using a gripper with mechanical variable stiffness to adapt to uncertainties and deal with error propagation; 3) exploiting the use of a hybrid gripper to execute the task using both pinch and suction grasps with the same end-effector, which benefits both the range of items that can be grasped and their placement in constrained spaces; and 4) We evaluate the methodology in a real robotic system using the DLR Hybrid Compliant Gripper (HCG) [3], integrating planning and insertion strategies into a full tight packaging pipeline. Various experiments are then carried out to test the effectiveness of the approach.

2 Problem Definition and Representation

To perform an efficient package planning, the dimensions of container and items are discretized using a given resolution. The container $C = [P_C^w, G_C]$, is described by the position P_C^w of its top-left corner in world coordinates. It is discretized with a planar grid $G_C = [res_C, g_C]$, characterized by the discretization resolution res_C, and the integer dimensions of the grid $g_C = [g_{C,i}, g_{C,j}]$. The same applies to the buffer $B = [P_B^w, G_B]$, which is represented as a 1D grid $G_B = [res_B, g_B]$.

The item set I is defined as a set of object-types. Each object $o_n = [\widetilde{P_{o_n}^w}, t_n, \{F_m\}_n]$ is an instance of one of the types available in I; it is characterized by its position in world frame $\widetilde{P_{o_n}^w}$, its type t_n, and by a set of faces $\{F_m\}_n$, which define its geometry. In turn, each face $\{F_m\}_n = [P_{F_{n,m}}, b_{n,m}, d_{n,m}]$ is characterized by the relative pose of its centre with respect to the centroid of the object $P_{F_{n,m}}$, a Boolean $b_{n,m}$ which defines if it can be grasped through suction, and the planar discretized dimensions $d_{n,m} = [d_i, d_j]$ that the item has when looked at from the normal along the surface.

The tight packaging task is defined as the collocation of the set of objects O into the container C; each item undergoes a cycle of grasp, planning and placement, as depicted in Fig. 2. In every cycle, a random object from the item set starts in a random pose in the grasp workspace; one object at a time is allowed to be in that region. The item is localized and grasped there. Once the operation is successful, the package planner is invoked to find a position for the item in the bin, using knowledge of the nominal state of the grid. An item can either find a placement, be assigned to the buffer, or be discarded.

In the first case, the planner outputs a target placement for the item in grid coordinates $P_{o_n}^g$, which corresponds to a target pose in world coordinates $P_{o_n}^w$.

[3] DLR HCG (2021) https://www.dlr.de/rm/en/desktopdefault.aspx/tabid-18061

The object placement is performed along a trajectory that the robot has to follow, defined as a set of poses $T = \{P_k\}$; each pose can be expressed as relative to the target position $P^w_{o_n}$, or in world coordinates, hence called respectively $T^t_{o_n} = \{P^t_k\}$ and $T^w_{o_n} = \{P^w_k\}$. The cycle is repeated until the container is filled, or an unrecoverable error happens. The container is considered filled if an item has been assigned to all the cells, and if the actual pose of each object placed $\widetilde{P^w_{o_n}}$ is within a threshold ϵ_a of the nominal one $P^w_{o_n}$. Similarly, an unrecoverable placement happens when the difference between actual and nominal pose is greater than another threshold ϵ_f.

3 Pipeline for tight packaging

Grasp and Manipulation approach. A prior analysis is performed to define the robot workspace for the tasks, using its capability map [14]. First, the *grasp workspace* is defined as a planar area of the table that the robot can reach and where the vision pipeline can return the poses of the items with sufficient confidence. Then, the best location for the container is found by checking that in the selected position the robot is able to pack items in the whole bin, collision-free and within its reachability. In a similar way, a region of the workspace is assigned to be the buffer.

Due to the constraints of tight packaging, a suction grasp is the preferred modality, as the gripper only needs access to one face of the item. On the contrary, with a pinch grasp a finger will always be in the way of a tight insertion. The grasp algorithm requires knowledge of the object type and its 6D pose, coming from a vision pipeline. With this, the algorithm computes which face of the item is looking upwards, and if it is possible to perform a suction grasp; if not, a pinch grasp is performed and the object is reoriented, in order to expose a different face. The pipeline for this process is described in Algorithm 1.

Package Planner and Item Selection. After a successful grasp, the planner (shown in Algorithm 2) selects where to place the item using as input the nominal state of the container, and the planar dimensions of the object $d_{n,m}$. First, the planner checks if the item should be buffered; this applies to small items, which can be useful in the latest stages of execution, where their size allows to fill in the irregular and small gaps remaining. This strategy is used until the buffer is full, or until it contains enough items to fill the remaining empty gaps. The Item Selection works in sync with this strategy: if the buffer needs to be emptied, the first item selected is the largest one in it; otherwise, the robot grasps the item laying in the grasp workspace. If the item is not buffered, the planner uses a Bottom-Left heuristic to find a placement for the item in the bin; the cost function implemented defines a *packing direction*, from the top-left corner towards the opposite one. If no placement is possible, the item is discarded.

Contact Analysis. Our packaging methodology relies on the detection of corners, defined by the edges of the items and the walls of the bin. A correctly oriented pushing action can move an item towards one corner, where it can fit guided by the constraints that it finds (Fig. 3-A). Moreover, as the bin fills up, the items constrain each other, reducing their freedom of movement.

Algorithm 1 Grasp and Manipulation

Input: Buffer B

$items\_number, \{\widetilde{P_{o_n}}, t_n\} \leftarrow call\_vision()$
if $items\_number \neq 1$ **then**
  return exit Error
$o_n \leftarrow select\_item(B, [\widetilde{P_{o_n}}, t_n])$
Create object: $o_n \leftarrow [\widetilde{P_{o_n}}, t_n, \{F_m\}_n]$
$F_{n,m} \leftarrow find\_upwards\_face(o_n)$
if $b_{n,m}$ **and not** *failure* **then**
  $success \leftarrow plan\_and\_execute\_suction\_grasp(o_n)$
  if *success* **then**
    failure ← False
    return exit ← Package Planner
  else
    failure ← True
    return exit ← Grasp and Manipulation
else
  $success \leftarrow plan\_and\_execute\_pinch\_grasp(o_n)$
  if *success* **then**
    $execute\_manipulation()$
    return exit ← Grasp and Manipulation
  else
    return exit ← Grasp and Manipulation

Insertion Strategy. By properly choosing a *loading direction* it is possible to keep piling objects in the container, correcting the arising uncertainties through a placing-by-pushing action. The insertion starts by selecting a loading direction depending on the current situation in the container (represented by the container grid), and the intended position of the item. As shown in Fig. 3-B, four directions are possible, defined as an angle in the container plane; each of them points from the center of the container towards one of the corners. The item is loaded preferentially towards the walls of the container; if not possible, against the packaging direction enforced by the planner. To insert the object in its slot, a set of actions is performed along the found direction, as depicted in Fig. 3-C. The item is first inclined backwards, to expose a corner, which is used to pierce between the items underneath; the held object is then pulled backwards, to clear some space, and straightened, pushing eventual items out of the way. Then, the item is moved forwards, onto its target position, and then lowered. This sequence of pushing and pulling does not disrupt the state of the bin achieved up to this moment. These actions are performed by moving the item along a trajectory defined by poses in space; these points are computed by a distance from the target position of the item in the container along the loading direction, a height from the container floor, and an orientation. Algorithm 3 details the execution.

Hybrid Grasp and Stiffness Level. To select a suitable hand configuration and stiffness level, the requirements of the task are analyzed, following our previous work in [10]. To accomplish the *insertion* it is necessary, on the one hand,

Algorithm 2 Package Planner

Input: o_n, C

if $o_n \in small\_items$ **and not** $full\_buffer$ **then**
  return $exit \leftarrow$ Buffer
$found$, $P^g_{o_n} \leftarrow find\_placement(d_{n,m}, G_C)$
if not found **then**
  $d_{n,m} \leftarrow [d_j, d_i]$
  $found$, $P^g_{o_n} \leftarrow find\_placement(d_{n,m}, G_C)$
if $found$ **then**
  return $exit \leftarrow$ Insertion
else
  return $exit \leftarrow$ Discard

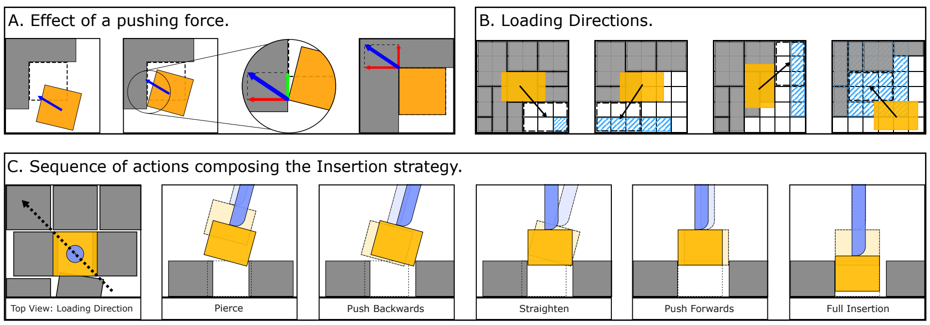

Fig. 3: A. Strategical selection of the pushing force drives the item toward a constraint and then complies into its target position. B. The four possible *loading directions*, depending on the target position (dashed). C. Sequence of actions composing the *insertion strategy* (loading from right to left in the side view).

to retain compliance in the horizontal plane, so to allow the item to look for contacts and to adapt to the surrounding despite the uncertainties. On the other hand, the robot needs to exert vertical forces to pierce and to insert the item, requiring a higher stiffness along said direction. The benefits of the hybrid gripper are here exploited: the item is grasped using suction, and the other fingertip is simply sitting on top of the item, to resist the vertical forces while allowing it to slide under its tip, thus keeping the required compliance (Fig. 4-A).

Corrective Actions. To minimize the chances of having objects occluding potential slots, it is possible to perform *corrective actions* after a placement. Given the actual and intended position of the item, a correction is triggered if their distance exceeds a given threshold. The correction is a pushing action along a direction, computed as a vector from $\widetilde{P^w_{o_n}}$ to $P^w_{o_n}$. It is performed by inserting a straight finger and moving it until a force exceeding a given threshold is sensed, signaling that a constraint has been found. Therefore, before executing the action, it is necessary to check if there is enough space to safely insert a finger. This is done considering the nominal situation in the container, verifying that

Algorithm 3 Insertion

Input: Bin C, item dimensions $d_{n,m}$, target pose $P^g_{o_n}$

$P^w_{o_n} = from\_relative\_to\_world\_coordinates(P^g_{o_n}, P^w_C)$
$\theta = compute\_loading\_direction(P^g_{o_n}, d_{m,n}, G_C)$
$T^w_{o_n} = compute\_trajectory(T^t_{o_n}, P^w_{o_n}, \theta)$
$execute\_trajectory(T^w_{o_n})$
return exit = Correction

a sufficient number of cells are empty in front of the item along the pushing direction. Algorithm 4 details the execution.

Algorithm 4 Correction

Input: Container grid G_C, target pose $P^w_{o_n}$, acceptance threshold ϵ_a

$\widetilde{P_{o_n}} = query\_item\_position()$
if $\widetilde{P^w_{o_n}} - P^w_{o_n} < \epsilon_a$ **then**
    return exit = Grasp and Manipulation
$\theta = compute\_correction\_direction(\widetilde{P^w_{o_n}}, P^w_{o_n})$
if $verify\_feasibility(G_C, P^w_{o_n}, \theta)$ **then**
    $plan\_and\_execute\_correction(P^w_{o_n}, \theta)$
return exit = Grasp and Manipulation

4 Experimental Setup

For the experimental task we use the redundant DLR LWR III with the mounted DLR Hybrid Compliant Gripper (HCG) and a F/T sensor placed between arm and gripper (Fig. 1). The HCG is a tendon-driven, underactuated, and hybrid gripper. Its fingers are the thumb modules of the CLASH hand [6], with 3 DoFs, slightly modified to accommodate a suction cup at the tip; there is an additional DoF at the base of each finger for enhancing the width of the grasp. The stiffness of each finger can be independently set, on a scale from 0 to 100% (the higher the value, the stiffer the finger), and independently of the finger configuration.

We capture RGB and depth images using an Azure Kinect DK camera. For computing the 3D position of an item, we first run GroundingDINO [9] to detect the item and pass the corresponding bounding box to a Segment Anything model [8] to extract the respective segmentation mask. Second, we derive the depth value of the segmentation mask's centroid and unproject the position with known camera intrinsics. Eventually, we return the position of the item with respect to the robot through a prior robot-to-camera calibration.

The packaging pipeline is implemented as a state machine, and relies on the DLR motion planning library *RMPL*. The DLR middleware *Links and Nodes* is used to enable communication among all the components. The user is involved in the execution, as the corrective actions are manually triggered, since this feedback is unavailable in the current implementation. The same setup is replicated in a simulation environment using Python, with PyBullet as physics engine.

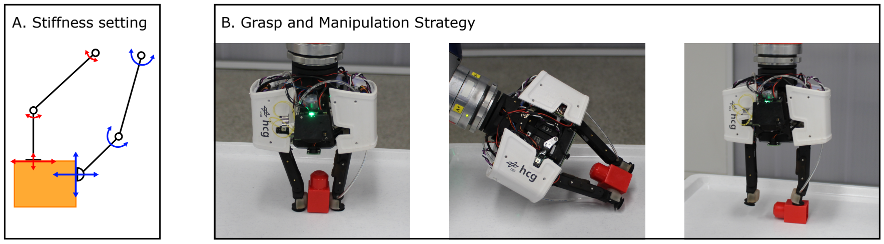

Fig. 4: A. HCG uses different stiffness levels in each finger: a suction cup with soft setting (blue) grasps the object while providing adaptability, and a fingertip with stiff setting (red) allows application of forces for insertion. B. A block being grasped and released in a pose that allows suction.

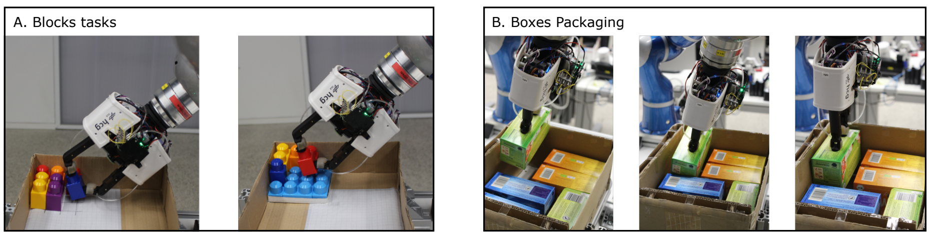

Fig. 5: A. Blocks packaging (left) and stacking (right). Note the hybrid grasp used for supporting the object. B. Frames of the insertion of a box.

5 Experimental evaluation: Results and Discussion

Three experiments are designed to test the proposed pipeline:

1. Blocks Packaging: Mega Blocks have to be packed upright in a half-opened container, both in simulation and reality, with nominal clearance between items of 1mm (Fig. 5-A). The blocks are graspable through suction only on the sides, with a grasp error of up to 15mm. Hence, this task emphasizes the grasp and manipulation aspect of the pipeline. A simplified insertion strategy is tested, since there is no need to incline the item and pierce. The hybrid grasp uses stiffness values of 10% and 40% of the maximum stiffness.
2. Blocks Stacking: follows the previous task, but now the blocks are stacked onto a base layer, rather than laying them on a flat surface (Fig. 5-A). This is performed in reality only, with a nominal clearance of 1mm, and fingers with stiffness of 10% and 60%.
3. Boxes Packaging: a closed container is to be filled with boxes, with a nominal clearance up to 15mm. The full insertion strategy is tested here (Fig. 5-B). In simulation, 5 shapes of boxes are used, and a buffer is present; in reality, the task is performed with tea boxes, with only one shape available. The grasp strategy uses a stiff finger (value equal to 30%) in the rear of the object, to pierce and push, and a soft one (value equal to 10%) in the front, to comply to the surrounding; the object is held with suction, due to its weight, with a grasp error that can be up to 25mm.

Table 1: Planner and buffer performance. Highlight: Settings for the experiments.

Grid Size	Itemset	Buffer	Spawned	Placed	Discarded	Leftovers
6x4	1x1 to 3x2	no	14.367 (179.62%)	9.217 (100.00%)	5.150 (79.62%)	-
6x4	1x1 to 3x2	yes	11.833 (147.90%)	8.437 (100.00%)	2.672 (40.96%)	0.724 (6.94%)
12x8	1x1 to 5x4	yes	39.345 (306.94%)	14.751 (100.00%)	22.307 (192.78%)	2.287 (14.17%)

Grasp and Manipulation The success rate for the grasp and manipulation portion of the pipeline is shown in Fig. 6-A. Finger grasping proves to be reliable and robust to errors. On the other hand, the main reason behind a failed Suction grasp is an imprecise object localization, and a lack of local dexterity in the robot; this is usually due to the robot working close to its joint limits. An unsuccessful manipulation is caused by either slippage, that lead to no toppling, or to the object falling in a pose that prevents its identification with the vision pipeline.

Package Planner To measure the performance of the planner, reported in Table 1, a batch of simulations is run with different container dimensions, size of the objects in the item set, and presence of a buffer. In each one, the number of items entering the system and their outcome is counted and averaged over the batch; similarly, the total area of the items for each outcome is summed and normalized with respect to the container area. The results show how the planner is able to run the task until completion, but having a large amount of items that end up *discarded*. The presence of a buffer improves the results, but performance drops drastically when increasing the variety in the item set.

Insertion Strategy: Blocks Packaging Fig. 6-B plots the overall success rate of the tasks; this is evaluated by measuring the maximum distance in which the obtained package exceeds the intended area of the container. Similarly, it is also possible to evaluate in which threshold does the error of each placement fall, and correlate the number of unsuccessful ones to the position in the grid. Such analysis shows that the error is less than 1cm (30% of the cell size) in 63.93% of the individual placements. Both simulation and reality show that the error is skewed along one direction. This is due to a lack of local dexterity, and at times, to the gripper getting in the way of a correct placement.

The presence of a grasp error is also correlated with the final placement; no significant difference is noticed, meaning that the pipeline is partially able to absorb such error, but also that its magnitude can be shadowed by the other error sources mentioned. Corrective actions are effective to reduce the misplacement inside the 5 mm threshold in 73.80% of the times. Repeating the experiment without them shows that the error increases at the end of the task, but not after each placement; this proves their necessity only for the containment of the outer layer, while the packaging strategy is able to deal with misplacements up to 1cm. The experiment is then repeated, first setting both fingers stiff, and then both soft. The results do not particularly worsen, but some behaviours can be

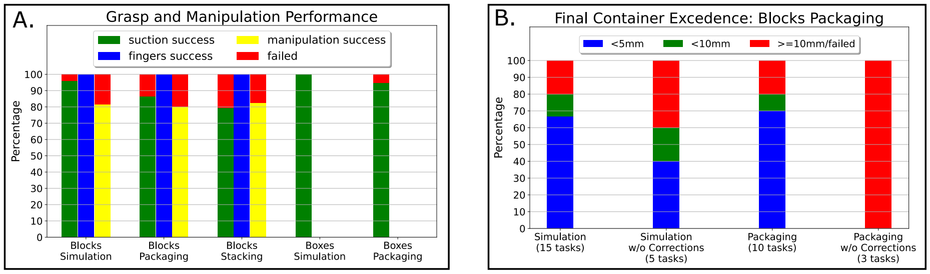

Fig. 6: A. Success ratio of the Grasp and Manipulation experiments. B. Success rate for blocks packaging.

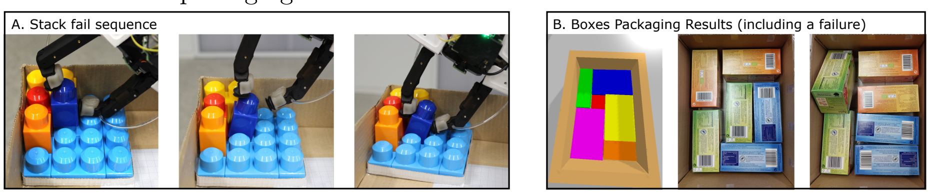

Fig. 7: A. Even after the block is released in the correct position, a tilted item produces failure at pressing down with the finger. B. Exemplary results in simulation and in experiments.

observed. First, the softer the hand, the more the fingers bend under the weight of the objects, approaching the target with residual inclination; moreover, the items tend to get more stuck against the container walls due to friction. With stiffer joints, higher forces are transmitted due to the higher rigidity of the system; in this way, an imprecise positioning of the item can squeeze another item out of its position, rather than comply to it.

Insertion Strategy: Blocks Stacking Along the 10 executions, the blocks are successfully placed in the right cell 92.3% of times; in the remaining, the item gets stuck on the wrong one, and leads to a failed task. In 47.9% of times, the block is pressed in its spot; in the remaining it is only released on it.

One more source of error is observed, which leads to these partial failures: the hand needs to release the block and reconfigure itself before pressing it in its position, to avoid having a finger colliding with the base-heads; in half of the executions, the hybrid grasp is not able to partially insert the block, which then tilts and is missed by the pressing finger (as shown in Fig. 7-A).

Insertion Strategy: Boxes Packaging The full insertion strategy is performed in these tasks; some examples of the outcomes are shown in Fig. 7-B.

Out of the 11 execution, 7 are successful, meaning that the container is filled completely. All the failures happen because the piercing action fails, meaning that the corner of the grasped box gets stuck onto the top of the ones underneath and is unable to enter between them, leading to the loss of the suction constraint. This happens because of the uncertainty in the relative position between object and robot, due to *grasp inaccuracy*. In presence of this error, even if the arm

Table 2: Summary of experimental results.

	Achieved clearance [mm] (% of item size)	**Extra space required**	**Success Rate**
Blocks Packaging	5 (16.1%)	yes	73.8%
Blocks Stacking	1 (3.2%)	yes	92.3%
Boxes Packaging	15 (10.3%)	**no**	63.3%

reaches the nominal position, the box does not, but is rather shifted back; hence the piercing action is not performed as intended and fails.

The task is then repeated using a simple insertion strategy, consisting in a top-down insertion of the item, directly in its intended position. This strategy fails 5 times out of 5, often before reaching the tight regions of the container; this happens because of the inability to deal with the large error coming from the grasping portion of the pipeline (up to 3 times the nominal clearance). The repetitions with low and high stiffness present a reduced success rate. The suction cups are quite compliant, hence their softness shadows the stiffness commanded to the fingers. In case of two soft fingers, the excessive compliance makes the execution even more vulnerable to getting stuck on the top of the boxes due to friction; making both fingers stiffer does not improve the behaviour, since only the rear finger actually utilizes its stiffness to pierce and push.

6 Conclusions

This paper presented a complete pipeline to do tight packaging. We focused on developing a strategy for inserting an item requiring only its intended area to be free; hence, our approach can fill a container up to all four walls. The use of a hybrid and variable-stiffness gripper ensures versatility while grasping, and allows suitable system behaviours to properly perform the item placement. There is scarce work so far on the tight packaging problem. The closest works in [15] and [20] use a gripper that prevents them from filling up the full container. However, our HCG and planning approach allow achieving this goal.

The results show an improvement in the ability to deal with uncertainty, with respect to a trivial strategy. Corrective actions are effective in reducing the errors, but are not necessary for smaller objects, as the insertion method can deal with them; moreover they cannot recover errors such as toppling and failed placements. The hybrid grasp and the stiffness settings prove useful in achieving the desired compliant behaviour necessary to perform each operation.

Still, improvements can be done in future work to further improve the efficacy of the proposed method. A fully trained visual pipeline could be used to localize the items, reducing the grasping uncertainty. The planner can be improved to search in multiple layers and in 3D; its efficacy would greatly improve with knowledge of the set of items. so that the planner can efficiently avoid discarding the items that do not fit in a given step. As a next step, we will complement the placement strategy with sensor feedback, so to find an actual contact by sensing it, in spite of possible shifts in the grasp. The corrections could be expanded to a set of recovery actions, so to tackle the different failures that can be encountered;

the detection of such failures could be automatized with visual feedback on the container, making the pipeline completely autonomous.

References

1. Berkey, J.O., Wang, P.Y.: Two-dimensional finite bin-packing algorithms. J. of the Operational Research Society **38** (1987)
2. Chen, V.K., Chin, L., Choi, J., Zhang, A., Rus, D.: Real- time grocery packing by integrating vision, tactile sensing, and soft fingers. IEEE RoboSoft (2024)
3. Cheng, X., Hou, Y., Mason, M.T.: Manipulation with suction cups using external contacts. In: Robotics Research. Springer Int. Pub., Cham (2022)
4. Correa, C., Mahler, J., Danielczuk, M., Goldberg, K.: Robust toppling for vacuum suction grasping. In: IEEE CASE (2019)
5. D'Avella, S., Sundaram, A.M., Friedl, W., Tripicchio, P., Roa, M.A.: Multimodal grasp planner for hybrid grippers in cluttered scenes. IEEE RA-L **8** (2023)
6. Friedl, W., Roa, M.A.: CLASH—A Compliant Sensorized Hand for Handling Delicate Objects. Frontiers in Robotics and AI **6**, 138 (2020)
7. Hong, Y.D., Kim, Y.J., Lee, K.B.: Smart pack: Online autonomous object-packing system using RGB-D sensor data. Sensors **20** (2020)
8. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. arXiv:2304.02643 (2023)
9. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv:2303.05499 (2023)
10. Martin, A.E.H., Sundaram, A.M., Friedl, W., Ruiz, V., Roa, M.A.: Task-oriented stiffness setting for a variable stiffness hand. In: IEEE ICRA (2023)
11. Morgan, A.S., Bateux, Q., Hao, M., Dollar, A.M.: Towards generalized robot assembly through compliance-enabled contact formations. IEEE ICRA (2023)
12. Morgan, A.S., Wen, B., Liang, J., Boularias, A., Dollar, A.M., Bekris, K.E.: Vision-driven compliant manipulation for reliable, high-precision assembly tasks. Robotic Science and Systems (RSS) (2021)
13. Pan, J.Y., Hui, K.H., Gao, X., Zhu, S., Liu, Y., Heng, P.A., Fu, C.W.: Sdf-pack: Towards compact bin packing with signed-distance-field minimization. IEEE/RSJ IROS pp. 10612–10619 (2023)
14. Porges, O., Stouraitis, T., Borst, C., Roa, M.A.: Reachability and Capability Analysis for Manipulation Tasks. In: ROBOT2013. Springer (2014)
15. Shome, R., Tang, W.N., Song, C., Mitash, C., Kourtev, H., Yu, J., Boularias, A., Bekris, K.E.: Tight robot packing in the real world: A complete manipulation pipeline with robust primitives (2019)
16. Su, J., Liu, C., Li, R.: Robot precision assembly combining with passive and active compliant motions. IEEE Trans. on Industrial Electronics (2021)
17. Wang, F., Hauser, K.: Robot packing with known items and nondeterministic arrival order. IEEE Trans. on Automation Science and Eng. **18** (2019)
18. Wang, F., Hauser, K.: Dense robotic packing of irregular and novel 3D objects. IEEE Trans. on Robotics **38** (2022)
19. Yang, S., Song, S., Chu, S., Song, R., Cheng, J., Li, Y., Zhang, W.: Heuristics integrated deep reinforcement learning for online 3D bin packing. IEEE Trans. on Automation Science and Engineering **21** (2024)
20. Zhou, Z., Zhang, Z., Xie, K., Zhu, X., Huang, H., Cao, Q.: A method of tight placement for robotic dense packing. ICARM (2022)