Multi-Object Grasping in the Plane

Wisdom C. Agboh[1,2], Jeffrey Ichnowski[2], Ken Goldberg[2], and Mehmet R. Dogar[1]

[1] School of Computing, University of Leeds, UK.
[2] The AUTOLab at UC Berkeley (automation.berkeley.edu), USA.

Abstract. We consider a novel problem where multiple rigid convex polygonal objects rest in randomly placed positions and orientations on a planar surface visible from an overhead camera. The objective is to efficiently grasp and transport all objects into a bin using multi-object push-grasps, where multiple objects are pushed together to facilitate multi-object grasping. We provide necessary conditions for frictionless multi-object push-grasps and apply these to filter inadmissible grasps in a novel multi-object grasp planner. We find that our planner is 19 times faster than a Mujoco simulator baseline. We also propose a picking algorithm that uses both single- and multi-object grasps to pick objects. In physical grasping experiments comparing performance with a single-object picking baseline, we find that the frictionless multi-object grasping system achieves 13.6% higher grasp success and is 59.9% faster, from 212 PPH to 340 PPH. See https://sites.google.com/view/multi-object-grasping for videos and code.

1   Introduction

There has been a recent increase in demand for fast and efficient robot picking systems, especially for warehouses [1, 2, 3, 4]. State-of-the-art robot grasping systems use only single-object grasps. As robot motion is increasingly becoming the bottleneck of pick-and-place systems [5, 6], reducing the number of arm motions with multi-object grasps has the potential to increase speed significantly. Furthermore, in cluttered scenes like Fig. 1, some of the objects may not have the necessary free space around them for the gripper to pick them individually in contrast to multi-object grasps.

Prior-work [7, 8, 9] studied multi-object grasps for the case where objects are already in contact and ready to be grasped. However, objects are oftentimes apart and
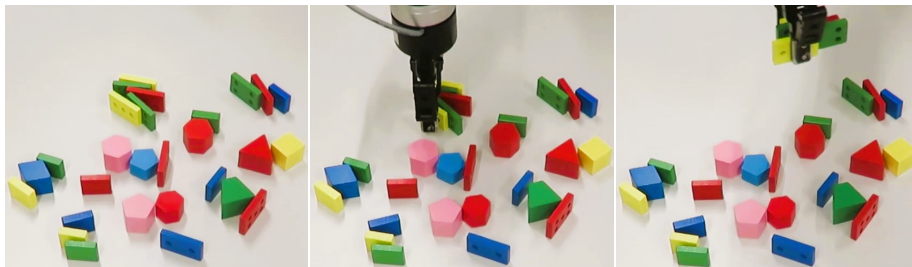

Fig. 1: Robot picks five objects at once with a multi-object push-grasp.

must be pushed/squeezed together before a multi-object grasp is possible. We call these *multi-object push-grasps*. They require physics-based predictions of how objects will move when the grippers close, e.g. to predict whether the objects will remain inside the grasp or whether they will slide out.

One approach to synthesizing such multi-object push-grasps is to divide it into two steps. Sakamoto et al. [10] proposed a system to pick two cuboids at once. Their approach pushes and aligns the objects first before grasping them. Such a two-step approach may not be feasible in cluttered scenes with more than two objects where object access is limited.

In this work, we propose a single-step multi-object push-grasp where objects are pushed and grasped simultaneously as the robot closes its gripper. One way to achieve that is to create models of objects in a physics simulator, generate single-step candidate grasps, and test them until a successful grasp is found. However, this is computationally expensive given the large number of physics simulations that could be required to find a grasp [11, 12, 13].

Therefore, we propose necessary conditions for multi-object push-grasping. We use these necessary conditions in a novel multi-object grasp planner to rank and filter grasp candidates, before testing them in a physics simulator. The result is a multi-object grasp planner that, on average, tests only two grasp candidates in a physics simulator before finding a successful one. In addition, we propose a picking system that uses the multi-object grasp planner to pick single or multiple objects at each time.

In simulation experiments, we find that our multi-object grasp planner is 19 times faster than a physics simulator baseline. In physical picking experiments compared to a single object picking baseline, we find that the multi-object grasping system achieves 13.6% higher grasp success, and is 59.9% faster. In summary, this work contributes:

1. Necessary conditions for frictionless multi-object push-grasping.
2. Theorems and proofs on the necessary conditions.
3. A multi-object grasp planner.
4. A picking system that uses both single and multi-object grasps.
5. Simulation experiments that compare our grasp planner with a physics simulator baseline and physical experiments that compare the picking system with a single object grasping baseline.

2   Related work

Picking multiple objects from a table or a bin is common for humans, e.g. waiters. Most prior work [2, 14, 15] in robotic picking focus on single object grasping. Recently, Sakamoto et al. [10] proposed a picking system that uses both one- and two-object grasps to pick cuboids. In this work, we propose a picking system that can grasp an arbitrary number of extruded convex polygonal objects at once.

A key question here is how to synthesize or generate grasps. There is a large body of work on synthesizing single-object grasps. They typically fall into one of two categories — analytic or data driven. The analytic methods [16, 17, 18] assume known object model and contact locations. They find grasps that can resist external wrenches or constrain the object's motion. On the other hand, data-driven approaches [19, 20, 21, 22, 23] generate grasping models that map directly from sensor readings like an RGB-D image to a successful grasp through various machine learning techniques.

Considering multi-object grasps, in several works [9, 24, 25], conditions for enveloping grasps of multiple objects using a multi-fingered robot hand, under the rolling contact assumption are proposed. Multiple objects (already in grasp wrench equilibrium) are grasped and lifted up towards the gripper's palm. Harada et al. [26] introduced active force closure for multiple objects. The work derives conditions to generate an arbitrary acceleration on multiple objects grasped with a multi-fingered robot hand. Yoshikawa et al. [27] proposed a condition to achieve power grasps where they formulate the multi-object optimal power grasp problem and minimize finger joint torques.

Other works [7, 8, 28, 29] propose methods to evaluate the grasp stability of multiple planar objects grasped by a multi-fingered robot hand. Fingers are replaced with 2D spring models and grasp stability is analysed through potential energy stored in the grasp. Yamada and Yamamoto [30] extended these multi-object grasp stability analyses to the 3D case, under both rolling and sliding contact. Results in these multi-object grasping works are mostly only numerical simulations, without real-robot multi-object grasps. In this paper, we derive conditions for equilibrium multi-object grasps, under the frictionless point-contact model. We also show real-robot multi-object grasps.

Recently, Chen et al. [31] investigated the problem of dipping a robot hand inside a pile of identical spherical objects, closing the hand and estimating the number of objects remaining in the hand after lifting. Shenoy et al. [32] also studied multi-object grasping in a similar setting of identical spherical objects, but with a goal of transferring the picked objects to another bin. Their approach generates a pre-grasp configuration and flexion strategy that corresponds to the desired number of objects to be picked from a given pile. Our work is focused on multi-object grasps of complex-shaped objects where physics predictions play a major role.

At the heart of various push-grasping methods [33, 34] is the need to make physics-based predictions of how objects will move when pushed and potentially squeezed together. It is computationally expensive to predict the result of these contact interactions [35, 36]. Thus, we propose necessary conditions for multi-object push-grasping for the first time to the best of our knowledge. We use these conditions in a novel multi-object grasp planner to synthesize multi-object grasps. Thereafter, we use our grasp planner in a novel picking system that generates both single and multi-object grasps and picks randomly placed objects.

3 Problem statement

As shown in Fig. 1, we consider the problem where multiple rigid convex polygonal extruded objects rest in randomly placed positions and orientations on a planar surface visible from an overhead camera. The goal is to develop a picking algorithm that efficiently grasps and transports all objects to a bin, using single and multi-object grasps.

Assumptions: We assume a frictionless point contact model only *between* objects in a multi-object grasp. We assume antipodal multi-object grasps where each object is kept in equilibrium by two neighbouring objects, or one object and a gripper jaw. We assume a parallel-jaw gripper and objects that are extruded convex polygons with uniform mass and known geometry.

State and action: The state $\mathbf{x}$ pose of all objects: $\mathbf{x} = [\{x_i, y_i, \theta_i\}]$, for $\mathrm{i} = 0, \ldots, N_o - 1$, where N_o is the number of objects on the table, x_i, y_i, and θ_i represent the 2D pose of object i, provided by an overhead camera. We represent single and multi-object grasp
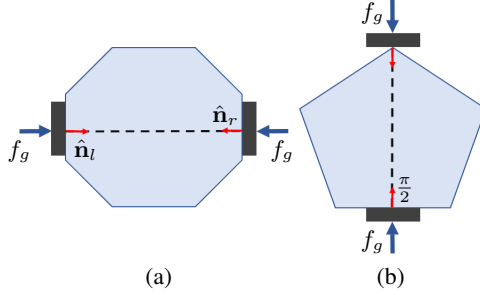
Fig. 2: Frictionless equilibrium grasps for single convex polygonal objects, with a parallel-jaw gripper. There are two possible grasp conditions in the frictionless case: (a) Two parallel edges. (b) Vertex and an opposing edge. The vertex has a perpendicular projection onto the edge.

actions in the same way: $\mathbf{u} = [x_g, y_g, \theta_g]$, where x_g, y_g, and θ_g represent the desired grasp pose of the gripper, after which the jaws close with a maximum force f_g. The grasp action involves four steps. (1) Moving the open gripper above the initial desired grasp pose, and lowering down until just above the surface/table. (2) Closing the gripper jaws. (3) Moving the gripper upwards, off the table, and above the bin. (4) Opening the jaws such that objects fall into the bin.

4   Multi-Object Grasps

To plan multi-object grasps, we need to understand conditions under which a multi-object grasp will succeed or fail. In this section, we study the conditions to obtain an equilibrium grasp for single and multiple planar, convex polygonal objects. Equilibrium grasps on an object occur when the grippers can apply forces and torques consistent with the contact model assumed at contact points, such that the net wrench on the object is zero. We study these grasps for multiple objects, under the frictionless point-contact model. In single-object grasping, under this contact model, the contact force from the left (l) and right (r) gripper plates can be written as: $\mathbf{f}_i = f_g \hat{\mathbf{n}}_i, \ i \in \{l, r\}$, where $\hat{\mathbf{n}}_i$ is the surface inward unit normal at the respective contact point, and f_g is the magnitude of a maximum gripper force. We seek equilibrium grasps under the frictionless point contact model. Thus, contact forces from the left and right grippers must satisfy:

$$f_g \hat{\mathbf{n}}_l + f_g \hat{\mathbf{n}}_r = \mathbf{0}, \quad \mathbf{p}_l \times f_g \hat{\mathbf{n}}_l + \mathbf{p}_r \times f_g \hat{\mathbf{n}}_r = \mathbf{0}, \tag{1}$$

where $\mathbf{p}_l$ and $\mathbf{p}_r$ are the position vectors of the left and right contact points respectively. Eq. 1 implies that for parallel-jaw grippers on a convex polygonal object, we can achieve equilibrium grasps only if the two contact normals have opposing directions ($\hat{\mathbf{n}}_l = -\hat{\mathbf{n}}_r$), and lie on the same line. These are antipodal grasps.

Antipodal grasps on a convex polygonal object occur under any of the following two necessary conditions: (i) two parallel edges as shown in Fig. 2a, and (ii) one vertex and an opposing edge where the perpendicular projection of the vertex lies inside the corresponding edge, as shown in Fig. 2b. There is a possible third case of grasping the convex polygonal object at two vertices, however that is unstable in the frictionless case and we do not consider such grasps.
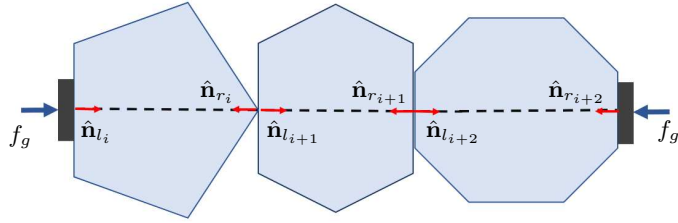
Fig. 3: A sample frictionless equilibrium grasp for multiple convex polygonal objects. Each object is in an antipodal grasp and all contact normals lie on the same line.

We extend the initial frictionless equilibrium grasp analysis to the multi-object case. To achieve an equilibrium grasp in the multi-object case, each object in the multi-object grasp must individually be in an antipodal grasp from contacts with grippers or neighbouring objects. In this work, we restrict our study to multi-object grasps where each object involved is kept in equilibrium by only two neighbouring objects or one object and a gripper plate. This implies that all contact normals across all objects must lie on the same line. We show a sample frictionless equilibrium multi-object grasp in Fig. 3.

Specifically, from the single object equation, Eq. 1, the multi-object equilibrium grasp equations for a set of n_o objects become:

$$f_g \hat{\mathbf{n}}_{l_i} + f_g \hat{\mathbf{n}}_{r_i} = \mathbf{0}, \quad i \in \{0, 1, \ldots, n_o - 1\} \tag{2}$$

$$\mathbf{p}_{l_i} \times f_g \hat{\mathbf{n}}_{l_i} + \mathbf{p}_{r_i} \times f_g \hat{\mathbf{n}}_{r_i} = \mathbf{0}, \quad i \in \{0, 1, \ldots, n_o - 1\}, \tag{3}$$

where $\hat{\mathbf{n}}_{l_i}$ and $\hat{\mathbf{n}}_{r_i}$ are the left and right contact normals for object i. For an equilibrium grasp, we find that $\hat{\mathbf{n}}_{l_i} = -\hat{\mathbf{n}}_{r_i}$, and both contact normals must lie on the same line for object $i \in \{0, 1, \ldots, n_o - 1\}$. This also implies that all normals across all n_o objects will lie on the same line.

5 Multi-Object Push Grasps

Objects almost never start in contact, fully aligned and ready to be grasped as shown in the previous section and in other prior work [25, 27, 30]. It is the case in practice that objects are usually apart and need to be squeezed together in a multi-object push-grasp. Hence, the need to make physics-based predictions of how objects will move as the robot pushes them together.

5.1 Two Necessary Conditions

Physics simulation for multi-object grasping is computationally expensive. We aim to avoid simulations and significantly speed-up multi-object grasp planning. In this section, we propose two necessary conditions for a multi-object push-grasp to succeed. We propose these conditions for the frictionless point-contact model. The conditions are: *multi-object diameter function* and *intersection area*. If any of these conditions is violated, the push-grasp is guaranteed to fail. We use these necessary conditions to filter out inadmissible grasps, and therefore to avoid computationally expensive simulations.
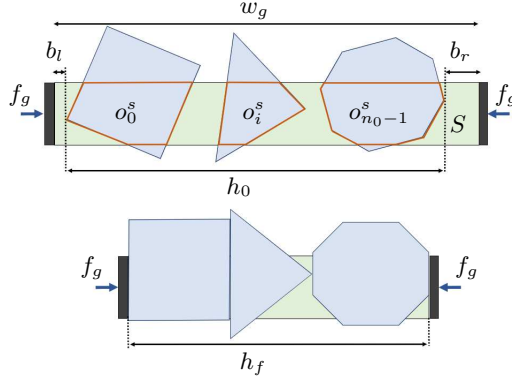
Fig. 4: A frictionless multi-object grasp. We show the initial internal rectangular region S, the gripper stroke w_g, the left and right minimum gripper distances to objects, b_l and b_r respectively, the initial multi-object grasp diameter (h_0), the final multi-object grasp diameter (h_f), and o_i^s, the intersection polygon (in red) between S and object O_i. We also show that $h_0 \geqslant h_f$ for any equilibrium multi-object grasp.

5.2   Multi-Object Diameter Function

In single-object grasping, consider quasi-statically closing a parallel-jaw gripper such that both jaws simultaneously make the first contact with the object. The distance between the jaws at this initial contact is called the *diameter*, first introduced by Goldberg [37] for frictionless grasping. Let this initial diameter be d_0. As we continue to close the grippers past this initial contact, the diameter must decrease until it reaches a local minimum value, at the stable grasp configuration. Let this final diameter be d_f. Thus, for any stable single-object grasp, we have that: $d_0 \geqslant d_f$. Next, we extend the single-object diameter condition to the multi-object case.

We begin by introducing the multi-object grasping diameter (h). We illustrate how to compute h given a grasp $\mathbf{u}$ in Fig. 4. We define the initial internal region between the parallel-jaw grippers to be the internal rectangular region, S. Let $o_i^s = S \cap O_i$, be the intersection polygon between S and object O_i. Let $w_g(t)$ be the gripper stroke at time t. Let $b_l(t)$ be the shortest distance between o_0^s and the left jaw, and $b_r(t)$ be the shortest distance between $o_{n_0-1}^s$ and the right jaw. Then, the multi-object grasp diameter is:

$$h(t) = w_g(t) - (b_l(t) + b_r(t)). \tag{4}$$

Let h_0 be the initial multi-object grasp diameter at time t_0, and let h_f be the corresponding final multi-object grasp diameter, at time t_f when the grippers become stationary after closing. $h_0 = h(t_0) = w_g(t_0) - (b_l(t_0) + b_r(t_0))$, and $h_f = h(t_f) = w_g(t_f) - (b_l(t_f) + b_r(t_f))$. When the grippers become stationary at t_f, $b_l(t_f) = b_r(t_f) = 0$, and $h_f = w_g(t_f)$. Since parallel-jaw grippers close in only one direction, it implies that $w_g(t_i) \geqslant w_g(t_{i+1})$, where $t_{i+1} > t_i$, and by extension $w_g(t_0) \geqslant w_g(t_f)$. Therefore, any multi-object grasp will satisfy:

$$h_0 \geqslant h_f \tag{5}$$

In Fig. 4, we show a sample multi-object initial diameter h_0 and final diameter h_f, for a frictionless multi-object grasp. The multi-object grasp diameter condition in Eq. 5
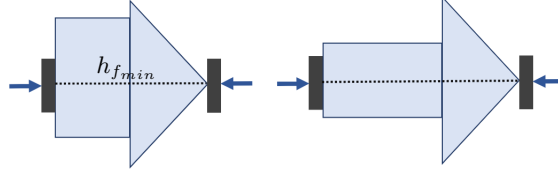
Fig. 5: All possible unique final frictionless multi-object grasp diameters for a rectangle and an isosceles triangle. There are only two unique multi-object grasp configurations (excluding vertex-vertex contacts). The left configuration has the minimum multi-object diameter $h_{f_{\min}}$.

requires knowledge of h_f which in turn requires a physics simulation to compute. This defeats the purpose of this condition.

Theorem 1. *Given a set of n_o objects, there exists a constant multi-object grasp diameter, $h_{f_{\min}}$ such that any grasp $\mathbf{u}$, with some initial multi-object grasp diameter h_0 is guaranteed to fail when $h_0 < h_{f_{\min}}$.*

Proof. Every polygonal object i in a successful multi-object grasp of n_o objects has a final diameter d_f^i, at time t_f when the grippers become stationary after they close. This corresponds to a multi-object grasp diameter $h_f = \sum_{i=0}^{n_0-1} d_f^i$. This summation is true for any successful multi-object grasp because contact normals across all n_o objects must lie on the same line (Eq. 2 and Eq. 3).

Every object has a minimum final diameter ($d_{f_{min}}^i$), at which a stable frictionless single-object grasp will occur. We can enumerate all final object grasp diameters for any convex polygonal object under the frictionless point-contact model using the two conditions shown in Fig. 2. Let the finite number of possible final diameters be $N_{d_f}^i$ for a given object i. The minimum final diameter can then be written as:

$$d_{f_{\min}}^i = \min(d_{f_j}^i), \quad j \in \{0, 1, \ldots, N_{d_f}^i - 1\}. \tag{6}$$

Then, we compute the minimum multi-object grasp diameter $h_{f_{\min}}$ at which a stable frictionless multi-object grasp of n_o objects will occur as:

$$h_{f_{\min}} = \sum_{i=0}^{n_o-1} d_{f_{min}}^i. \tag{7}$$

Recall from Eq. 5 that $h_0 \geqslant h_f$ for any multi-object grasp, but $d_f^i \geqslant d_{f_{\min}}^i$ which implies $h_f \geqslant h_{f_{\min}}$. Therefore any successful multi-object grasp must satisfy:

$$h_0 \geqslant h_{f_{\min}}. \tag{8}$$

Thus, if the initial multi-object grasp diameter is less than the minimum possible final multi-object grasp diameter ($h_0 < h_{f_{\min}}$), the grasp $\mathbf{u}$ is guaranteed to fail. This means that h_0 is so small that as the gripper closes, a stable multi-object grasp diameter can never be reached, completing the proof.   □

For example, consider the case in Fig. 5, where we have two objects such that $n_o = 2$. The rectangular object (O_0) has two unique stable grasp configurations under frictionless grasping with a parallel-jaw gripper ($N_{d_f}^0 = 2$), disregarding the other configurations due to symmetry, and vertex-vertex grasps. Similarly, the triangular object (O_1) has only one unique stable grasp configuration ($N_{d_f}^1 = 1$), again disregarding vertex-vertex grasps. Thus, there are two unique multi-object grasp configurations in the frictionless case and the left configuration has the minimum multi-object grasp diameter $h_{f_{\min}}$. We show an example of a grasp that failed due to a violation of the multi-object grasp diameter condition in Fig. 7(top).

5.3   Intersection Area

Each object in a multi-object equilibrium grasp needs to also be in a single-object equilibrium grasp. This requires at least two contacts, whether from the parallel jaws or other objects. Thus, as the gripper closes, if we predict no contacts for at least one object, the multi-object grasp fails.

Let $A_i(t) = \mathrm{Area}(o_i^s)$, be the area of the intersection polygon for object i, during a multi-object grasp. Then, we require the intersection area for each object be greater than zero when the gripper closes. $A_i(t)$ changes as the gripper closes with constant force f_g, and hence is a function of time. Thus, the intersection area necessary condition for n_o objects become:

$$A_i(t_f) > 0, \ i \in \{0, 1, \ldots, n_o - 1\} \tag{9}$$

where $t_f > 0$ is when the gripper becomes stationary after closing. Knowing exactly how $A_i(t)$ changes as the gripper closes requires a physics simulation, which defeats the purpose of this condition. However, observe that if the starting intersection area for an object is zero it is likely to remain so as the gripper closes. This is especially true when the object of interest is sufficiently far away from S.

Theorem 2. *Given an object O_i in a multi-object grasp $\mathbf{u}$ of n_o objects, if $A_i(0) = 0$ then there exists some minimum distance $\epsilon > 0$ between O_i and S such that the multi-object grasp is guaranteed to fail if the distance between O_i and S is larger than ϵ.*

Proof. Consider Fig. 6. Object O_i moves towards S and can subsequently achieve $A_i(t_f) > 0$ only when there is contact between it and other objects as the gripper closes. ϵ is the minimum distance between O_i and S where surrounding objects cannot make contact as the gripper closes with grasp $\mathbf{u}$. Let m_j for a surrounding object j be the length of its longest diagonal, and let $M = \sum_{j=1}^{n_o-1} m_j$ be the sum of such diagonals for all other objects. This is the farthest other $(n_o - 1)$ objects can be lined up along the normal direction ($\hat{\mathbf{n}}_S$) of S to reach O_i. Assuming quasi-static grasping, other objects cannot make contact with O_i beyond M. It is an upper bound on ϵ such that $\epsilon \leqslant M$. Therefore if O_i is at least M minimum distance away from the internal rectangular region and $A_i(0) = 0$, the multi-object grasp, including the object O_i is guaranteed to fail, which completes the proof.   □

In practice, M would be too large to provide useful grasp filtering. Recall that there is a minimum grasp diameter ($h_{f_{\min}}$) at which a multi-object grasp succeeds. This means that starting at h_0, the grippers can only move a maximum distance of $\Delta h_{\max} = h_0 -$
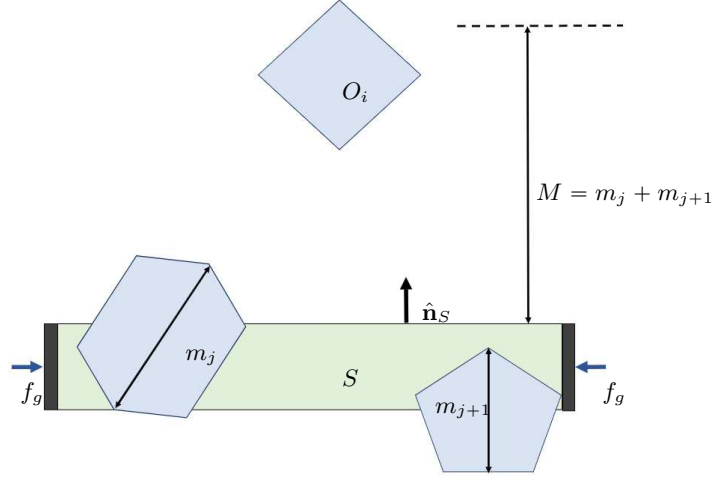
Fig. 6: An illustration of the intersection area necessary condition's proof. $M = \sum_{j=1}^{n_o-1} m_j$ is an upper bound on ϵ, the minimum distance between O_i and S such that the multi-object grasp is guaranteed to fail when $A_i(0) = 0$.

$h_{f_{\min}}$, otherwise the grasp fails. Under quasi-static grasping and given that the motion of objects in S are only due to the gripper motion, we assume each surrounding object will move at most by $\Delta h_{\max}$ along $\hat{\mathbf{n}}_S$ and towards O_i. Given, that $\Delta h_{\max}$ is small for a standard parallel-jaw gripper, we assume that contacts from surrounding objects O_j will be too short to bring object O_i into the internal rectangular region S. Therefore, we approximate $\epsilon = 0$, such that the relaxed intersection area condition becomes:

$$A_i(0) > 0, \ i \in \{0, 1, \dots, n_o - 1\}. \tag{10}$$

For example, take Fig. 6 where the multi-object grasp diameter condition is satisfied. Since $A_i(0) = 0$, the grasp is guaranteed to fail with the relaxed condition in Eq. 10. We show an example of a multi-object grasp that failed due to a violation of the intersection area necessary condition in Fig. 7 (bottom).

6 Picking System

This section details a picking system that uses both single and multi-object grasps to pick randomly placed objects.

6.1 Picking Algorithm

We present a picking algorithm in Alg. 1. Given a current state $\mathbf{x}$ containing N_o objects (line 2), the algorithm finds a multi-object grasp for the largest group of objects that can be grasped together, using the GraspPlanner(.) subroutine (line 6) and executes the grasp, until no objects are left on the table or a time limit is reached (line 10). The CreateObjGroups(.) subroutine (line 3) loops through center points of objects and creates groups of all objects whose centers are within half a gripper width radius. Thereafter
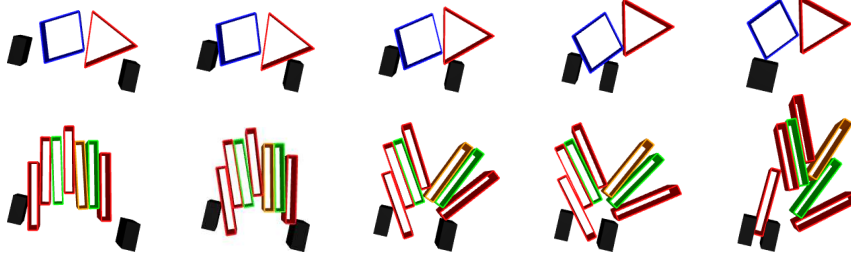
Fig. 7: Two examples of frictionless multi-object grasps that failed due to a violation of the two conditions. **Top row**: Multi-object diameter failure. **Bottom row**: intersection area failure. The black rectangles are the parallel jaws of the gripper.

it eliminates all object groups that have a superset in the object groups list, apart from the single objects. The RankObjGroups(.) subroutine (line 4) ranks the list of object groups by their size, such that the grasp planner attempts to find a grasp for the largest object groups first. We provide details of the GraspPlanner(.) subroutine in the subsequent section. A grasp execution can result in failure where desired objects escape the grippers and other objects can be displaced from their original poses. The online nature of Alg. 1, permits finding new grasps until the table is cleared or a time limit is reached.

6.2   Frictionless Multi-Object Grasp Planner (GP)

In this section, we propose a multi-object grasp planner. It is summarized in Alg. 2. The GraspPlanner(.) returns a grasp $\mathbf{u}$ if it can find one for the current group of objects, obj_group, given the current state.

First, the algorithm generates multiple grasp candidates (line 1). Second, it ranks these grasp candidates based on their likelihood of being successful (line 2). Finally, we sequentially test grasps to find a successful one (lines 3–6). To test a multi-object grasp we first use a grasp filtering system. We check if the grasp satisfies the two neccessary conditions outlined in Sec. 5 on line 4. If the necessary conditions are satisfied, we check for grasp success using a physics simulator (line 5). We explain the main subroutines in the following paragrasphs:

GenGraspCands( ): Recall that a grasp action is parametrized as the position and orientation of the gripper: $\mathbf{u} = [x_g, y_g, \theta_g]$. The gripper reaches the desired pose and

---

Algorithm 1: Picking Algorithm

1 do
2 $\quad$ $\mathbf{x}$, $N_o \leftarrow$ GetCurrentState(.)
3 $\quad$ obj_groups $\leftarrow$ CreateObjGroups($\mathbf{x}$)
4 $\quad$ ranked_obj_groups $\leftarrow$ RankObjGroups(obj_groups)
5 $\quad$ **for** obj_group **in** ranked_obj_groups **do**
6 $\quad\quad$ $\mathbf{u} \leftarrow$ GraspPlanner($\mathbf{x}$, obj_group)
7 $\quad\quad$ **if** $\mathbf{u} \mathrel{!{=}} \{\}$ **then**
8 $\quad\quad\quad$ Execute $\mathbf{u}$
9 $\quad\quad\quad$ break
10 **while** $N_o > 0$ **and** time remaining;

Algorithm 2: Grasp Planner (GP)

    Input   : **x**: Current state, obj_group: Objects in the potential grasp
    Output   : **u**: A grasp action
1 grasp_cands ← GenGraspCands(**x**, obj_group)
2 ranked_grasp_cands ← RankGraspCands(grasp_cands)
3 **for u in** ranked_grasp_cands **do**
4   **if not** GraspFailure*(**x**, **u**)* **then**
5   **if** CheckGraspSuccess*(**x**, **u**)* **then**
6   **return u**
7 **return** {}

---

closes. Given a group of objects, we first find the convex hull of all of the objects. Then, we generate N_p position samples that uniformly cover the convex hull. At each position sample, we generate N_θ orientation samples. We reject samples that result in collisions, between the jaws and any object — inside or outside the group.

RankGraspCands( ): We aim to find a parameter that quickly predicts the likelihood of a multi-object push-grasp success and use it to rank grasp candidates. Recall the two necessary conditions for multi-object push-grasp success in Sec. 5 — multi-object diameter and intersection area. These conditions suggest that the larger the area of the intersection polygon for each object, the more likely a grasp will be successful. Therefore, we propose to use the total intersection area ($A_T = \sum_{i=0}^{n_o-1} A_i$) as a heuristic to judge how successful a grasp will be. The higher the total area, the more likely the grasp will be successful. A_i is computed as the area of intersection between object O_i and S.

GraspFailure( ): We filter multi-object grasps using the two conditions provided in Sec. 5 — multi-object grasp diameter and intersection area. If any of the two necessary conditions is not satisfied, the multi-object grasp is guaranteed to fail.

CheckGraspSuccess( ): To check the success of a grasp **u**, we use a physics simulator where we assume objects are rigid, and have a uniform mass distribution. We close the gripper at the specified pose, then lift it above the table to check that the grasp is stable. Using a physics simulator means that our grasp planning system has no false positives for simulation experiments. However, we expect some grasps to fail in physical experiments due to uncertainty in state and model parameters and aleatory uncertainty related to friction, sensing, and control.

7 Experiments

We conduct simulation and physical experiments to evaluate the multi-object push-grasping necessary conditions, grasp planner, and picking system. In the following subsections, we explain the general setup, experimental details, baselines, and results.

7.1 Simulation and Physical Experiments:

We use Mujoco 2.1.0 as the physics simulator. In all experiments, we use a fixed set of objects, shown in Fig. 1. It contains a total of 33 objects from 3-sided to 6-sided convex polygons. We create a model of these objects in Mujoco where we ensure that objects are rigid and have a uniform mass distribution. In physical experiments, we use an RGB-D camera to estimate the pose of objects on a table. We approximate the frictionless case by selecting low ($\mu = 0.01$) friction parameters in Mujoco.
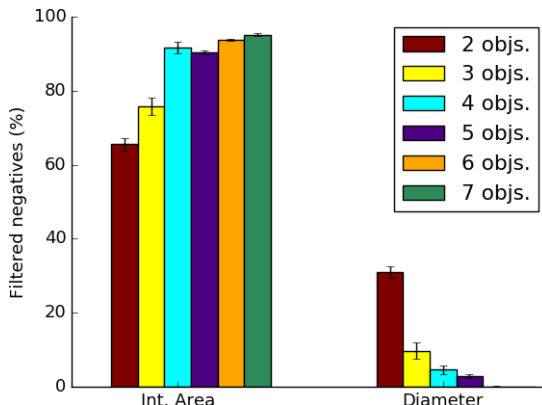
Fig. 8: Simulation experimental evaluation of the necessary conditions for multi-object push-grasping. Using both the intersection area (Int. Area) and the multi-object diameter (Diameter) necessary conditions, we can predict on average across all objects **93.4%** of true-negative grasps in any given set of grasp candidates. The average number of grasp candidates in any given scene is **193.19 ± 1.54**. All errors are within 95% confidence interval of the mean.

7.2   Evaluation of Necessary Conditions in Simulation:

In this section, we conduct simulation experiments to evaluate the proposed necessary conditions for multi-object push-grasping. We use a physics simulator to get ground-truth multi-object push-grasping results. If a necessary condition is violated, the grasp is predicted to fail (negative). If the ground truth also fails, we term this a true negative, otherwise it is a false negative. An important question here is what percentage of true negatives can we predict using these necessary conditions? Are there any false negatives? These questions will specifically help us evaluate the practicality of our intersection area assumption in Sec. 5.

We consider classes of 2-object to 7-object grasps (limited only by gripper width). For each multi-object grasp class, we create 200 randomly generated scenes. That is a total of 1200 grasping scenes. Thereafter, for each grasp scene, we generate $N_p \times N_\theta = 70 \times 7$ grasp candidate samples, and reject samples in collision. For each grasp candidate, we check both multi-object push-grasping necessary conditions and compare results to a Mujoco simulation of the grasp. We record any true negatives and any false negatives, across all grasp candidates, grasp scenes, and classes.

Results on evaluating the necessary multi-object push-grasping conditions can be found in Fig. 8. We did not record any false negatives across any object class—a result that supports the mild intersection area assumption. A majority of the true-negative grasps (**85.4%** on average) were predicted by the intersection area condition, while the multi-object diameter function predicted **8%**. Both conditions combined predicted **93.4%** of true negatives, with a **0.5%** overlap between their predictions only for the 2-object case. We find that the multi-object grasp diameter condition violations reduce with increasing number of objects (from **31%** for two objects), while violations for the intersection area increased. See Fig. 7 for grasps that failed due to a violation of the multi-object diameter condition and the intersection area condition.

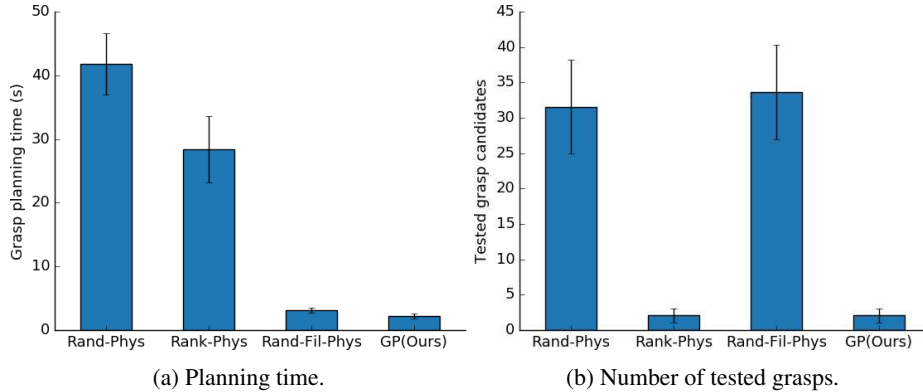(a) Planning time.   (b) Number of tested grasps.

Fig. 9: Experimental results in simulation on multi-object grasp planning. Our approach (GP) found a grasp or reported that none exists in **2.2 seconds** on average which is **19 times faster** than the Mujoco physics simulator baseline (Rand-Phys), thanks to a fast grasp filtering system. GP tests an average of 2 candidate grasps before a successful grasp is found. That is **16.2 times less** candidate grasps using the ranking system, compared to a random approach (baselines Rand-Phys, and Rand-Fil-Phys). Error bars are shown within 95% confidence interval of the mean.

7.3   Grasp Planning Experiments in Simulation

We conduct experiments in simulation to evaluate the multi-object grasp planner. Our goal here is to understand the effects of the grasp filtering and ranking systems. We follow the same process as Sec. 7.2 to generate random scenes and corresponding grasp candidates. In each scene, we use the grasp planner (GP) and three baselines to find a grasp. We record the grasp planning time — total time to return a grasp or report that no successful grasp was found. We also record the number of grasp samples tested/considered before a grasp is found.

Baseline 1 (Rand-Phys): This is a version of the multi-object grasp planner where we randomize the grasp candidates after they are generated (no grasp ranking). Then test each grasp sequentially in only the physics simulator (no grasp filtering), until a grasp is found or all grasp candidates are tested.

Baseline 2 (Rank-Phys): This is a version of the multi-object grasp planner that ranks grasps but tests grasps sequentially using only the physics simulator (no grasp filtering).

Baseline 3 (Rand-Fil-Phys): This randomizes the grasp candidates (no ranking) as in baseline 1, but uses grasp filtering and the physics simulator.

Results: We show simulation multi-object grasp planning experimental results in Fig. 9. Planning time results are in Fig. 9a, results on grasps tested in the physics simulator are in Fig. 9b. Our proposed approach (GP) has the lowest planning time of **2.2 seconds** on average which is **19 times faster** than directly using only a physics simulator (Rand-Phys). We note that the physics simulator models the full dynamics of the system but we do not, and this explains the speedup. We find that the grasp ranking method reduced the number of tested grasps by **93.44%**, when we compare Rand-Phys and Rank-Phys.

Table 1: Physical picking experimental results with 20 scenarios, each with 33 objects, placed in different random positions and orientations.

Methods	Success rate (%)	Percent picked	PPH
Single-Object	73.31 ± 4.84	97.77 ± 1.42	212.59 ± 19.03
Multi-Object	83.33 ± 4.29	99.52 ± 0.48	340.08 ± 27.09

7.4 Physical Experiments

We conduct physical picking experiments to evaluate the picking system presented in this work, in comparison to a single object picking system. Details of this single-object picking system can be found below.

Single object picking system: We use the same multi-object picking algorithm and grasp planner for single object picking. However, we restrict the number of objects to a randomly chosen single object.

Metrics: We compare both picking systems based on success rate, percentage of objects picked, and picks per hour. *Success rate*: the percentage of grasp attempts that moved at least one object into the box. *Percent picked*: the fraction of objects that were moved to the box. *Picks per hour (PPH)*: total number of objects picked per hour, through single or multi-object grasps.

Physical experiment details: We use the UR5 robot arm with a Robotiq 2F-85 gripper, and an Astra RGBD camera in all experiments. We create 20 picking scenes where we randomly place 33 objects in different positions and orientations. We clustered objects arbitrarily to create opportunities for multi-object grasps. In each scene, we use both the multi-object picking system and the single-object picking system to generate grasps. We replicated the scene manually between the two picking systems, leading to a total of 40 real-robot runs. Please see Fig. 1 for a sample real-world scene. A failed grasp attempt is where the robot misses a grasp (all objects escape), or where all objects fall out of the gripper before they reach the box.

Results: Picking results can be found in Table. 1. Both systems picked a similar percentage of objects. However, the multi-object grasping system achieved **13.6%** higher grasp success. A major source of grasp failure is uncertainty in state and model parameters. This affects success rates of both picking systems. The single-object system suffers more as it makes more grasp attempts aimed at picking single objects that are oftentimes not easily accessible due to clutter. Another source of grasp failure is model mismatch. Grasps were planned in the physics simulator and the underlying assumptions there may not hold on the real physical system. We find that the multi-object grasping system is **59.9%** faster than the single object picking baseline in picks per hour (PPH). This is in spite of the fact that uncertainty can sometimes aid the single-object picking system in grasping more than one object at a time.

8   Discussion and future work

We propose the planar frictionless multi-object grasping problem where multiple convex polygonal objects are grasped and transported to a bin. We provide necessary conditions for multi-object push grasping under the frictionless point contact model, and their corresponding theorems and proofs. Experiments in simulation and on the physical robot suggest a significant speed-up in grasp planning time compared to a physics

simulator baseline. We also find a reduction in overall picking time, compared to a single-object picking system.

One assumption in this work is the frictionless point contact model between objects. This limits the number of feasible multi-object grasps. In future work, we will consider friction and how it affects multi-object grasps. We planned grasps in a physics simulator and without considering uncertainty. This lead to grasp failure. We will extend this work to generate robust multi-object grasps and to consider general 3D objects, including deformables. We restricted our study to antipodal multi-object grasps. In future work, we will consider other forms of multi-object grasps, including multi-fingered grasps.

9 Acknowledgement

Bibliography

[1] Michael Danielczuk, Jeffrey Mahler, Chris Correa, and Ken Goldberg. Linear Push Policies to Increase Grasp Access for Robot Bin Picking. In *CASE*, 2018.

[2] Jeffrey Mahler and Ken Goldberg. Learning deep policies for robot bin picking by simulating robust grasping sequences. In *CoRL*, volume 78, pages 515–524, 2017.

[3] Ryo Matsumura, Yukiyasu Domae, Weiwei Wan, and Kensuke Harada. Learning based robotic bin-picking for potentially tangled objects. In *IROS*, 2019.

[4] Huang Huang, Marcus Dominguez-Kuhne, Vishal Satish, Michael Danielczuk, Kate Sanders, Jeff Ichnowski, Andrew Lee, Anelia Angelova, Vincent Olivier Vanhoucke, and Ken Goldberg. Mechanical Search on Shelves using LAX-RAY: Lateral Access X-RAY. In *IROS*, 2021.

[5] Jeffrey Ichnowski, Michael Danielczuk, Jingyi Xu, Vishal Satish, and Ken Goldberg. Gomp: Grasp-optimized motion planning for bin picking. In *ICRA*, 2020.

[6] Jeffrey Ichnowski, Yahav Avigal, Vishal Satish, and Ken Goldberg. Deep learning can accelerate grasp-optimized motion planning. *Science Robotics*, 5(48), 2020.

[7] Takayoshi Yamada, Shuichi Yamanaka, Manabu Yamada, Yasuyuki Funahashi, and Hidehiko Yamamoto. Grasp stability analysis of multiple planar objects. In *ROBIO*, 2009.

[8] Takayoshi Yamada, Manabu Yamada, and Hidehiko Yamamoto. Stability analysis of multiple objects grasped by multifingered hands with revolute joints in 2d. In *IEEE International Conference on Mechatronics and Automation*, 2012.

[9] K. Harada and M. Kaneko. Kinematics and internal force in grasping multiple objects. In *IROS*, 1998.

[10] Takumi Sakamoto, Weiwei Wan, Takao Nishi, and Kensuke Harada. Efficient picking by considering simultaneous two-object grasping. In *IROS*, 2021.

[11] Wisdom C. Agboh, Daniel Ruprecht, and Mehmet. R. Dogar. Combining coarse and fine physics for manipulation using parallel-in-time integration. *ISRR*, 2019.

[12] Wisdom C. Agboh, Daniel Ruprecht, and Mehmet R. Dogar. Parareal with a learned coarse model for robotic manipulation. *Comput. Visual Sci.*, 23(8), 2020.

[13] Mohamed Hasan, Matthew Warburton, Wisdom C. Agboh, Mehmet R. Dogar, Matteo Leonetti, He Wang, Faisal Mushtaq, Mark Mon-Williams, and Anthony G. Cohn. Human-like planning for reaching in cluttered environments. In *ICRA*, 2020.

[14] Douglas Morrison, Peter Corke, and Jürgen Leitner. Learning robust, real-time, reactive robotic grasping. *IJRR*, 39(2-3):183–201, 2020.

[15] Xibai Lou, Yang Yang, and Changhyun Choi. Collision-aware target-driven object grasping in constrained environments. In *ICRA*, 2021.

[16] Domenico Prattichizzo and Jeffrey C. Trinkle. Grasping. In *Springer Handbook of Robotics*, pages 671–700, 2008.

[17] Alberto Rodriguez, Matthew T Mason, and Steve Ferry. From caging to grasping. *IJRR*, 31 (7):886–900, 2012.

[18] Ben Kehoe, Akihiro Matsukawa, Sal Candido, James Kuffner, and Ken Goldberg. Cloud-based robot grasping with the google object recognition engine. In *ICRA*, 2013.

[19] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.

[20] Corey Goldfeder and Peter K. Allen. Data-driven grasping. *Autonomous Robots*, 31:1–20, 2011.

[21] Jeffrey Mahler, Florian T Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kröger, James Kuffner, and Ken Goldberg. Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards. In *ICRA*, 2016.

[22] Leo Pauly, Wisdom C Agboh, David C Hogg, and Raul Fuentes. O2a: One-shot observational learning with action vectors. *Frontiers in Robotics and AI*, 8, 2021.

[23] Wissam Bejjani, Wisdom C. Agboh, Mehmet R. Dogar, and Matteo Leonetti. Occlusion-aware search for object retrieval in clutter. In *IEEE IROS*, 2021.

[24] K. Harada and M. Kaneko. Enveloping grasp for multiple objects. In *ICRA*, 1998.

[25] K. Harada, M. Kaneko, and T. Tsujii. Rolling-based manipulation for multiple objects. *IEEE Transactions on Robotics and Automation*, 16(5):457–468, 2000.

[26] K. Harada, M. Kaneko, and T. Tsuji. Active force closure for multiple objects. In *Advances in Robot Kinematics*, pages 155–164, 2000.

[27] T. Yoshikawa, T. Watanabe, and M. Daito. Optimization of power grasps for multiple objects. In *ICRA*, 2001.

[28] T. Yamada, T. Ooba, T. Yamamoto, N. Mimura, and Y. Funahashi. Grasp stability analysis of two objects in two dimensions. In *ICRA*, 2005.

[29] T. Yamada, N. Mimura, and Y. Funahashi. Grasp stability analysis of two objects with both friction and frictionless contacts in two dimensions. In *IEEE International Symposium on Micro-NanoMechatronics and Human Science*, 2005.

[30] Takayoshi Yamada and Hidehiko Yamamoto. Static grasp stability analysis of multiple spatial objects. *Journal of Control Science and Engineering*, 3:118–139, 2015.

[31] Tianze Chen, Adheesh Shenoy, Anzhelika Kolinko, Syed Shah, and Yu Sun. Multi-object grasping - estimating the number of objects in a robotic grasp. In *IROS*, 2021.

[32] Adheesh Shenoy, Tianze Chen, and Yu Sun. Multi-object grasping - generating efficient robotic picking and transferring policy. *CoRR*, 2021.

[33] Mehmet Dogar and Siddhartha S. Srinivasa. A framework for push-grasping in clutter. In *RSS*, 2011.

[34] Wisdom C. Agboh and Mehmet R. Dogar. Real-time online re-planning for grasping under clutter and uncertainty. In *IEEE Humanoids*, pages 1–8, 2018.

[35] Wisdom C. Agboh and Mehmet R. Dogar. Pushing fast and slow: Task-adaptive planning for non-prehensile manipulation under uncertainty. In *WAFR*, 2018.

[36] Wisdom C. Agboh and Mehmet Remzi Dogar. Robust physics-based manipulation by interleaving open and closed-loop execution. *CoRR*, abs/2105.08325, 2021.

[37] Kenneth Y. Goldberg. Orienting polygonal parts without sensors. *Algorithmica*, 10:201–225, 1993.