

# Discrete Bimanual Manipulation for Wrench Balancing

Silvia Cruciani<sup>\*1</sup>

Diogo Almeida<sup>\*1</sup>

Danica Kragic<sup>1</sup>

Yiannis Karayiannidis<sup>2</sup>

**Abstract**—Dual-arm robots can overcome grasping force and payload limitations of a single arm by jointly grasping an object. However, if the distribution of mass of the grasped object is not even, each arm will experience different wrenches that can exceed its payload limits. In this work, we consider the problem of balancing the wrenches experienced by a dual-arm robot grasping a rigid tray. The distribution of wrenches among the robot arms changes due to objects being placed on the tray. We present an approach to reduce the wrench imbalance among arms through discrete bimanual manipulation. Our approach is based on sequential sliding motions of the grasp points on the surface of the object, to attain a more balanced configuration. We validate our modeling approach and system design through a set of robot experiments.

## I. INTRODUCTION

Robots can be intrinsically safe to enable cooperation with humans. However, this safety comes at the cost of limited payload for the robot’s arms. Dual-arm robotic systems are able to overcome individual arm’s payload limitations by jointly grasping an object. A higher degree of stability is possible while exerting a bimanual grasp [1]–[3], in face of external forces.

In this paper, we address the problem of shifting a dual-arm robot’s grasp points on a tray to compensate for variations on the tray’s load (see the robot in Fig. 1 for an example). We balance the force and torque load between the two arms to maintain the tray and the objects on it in equilibrium. Whenever an object is added or removed from the tray, its center of mass (COM) will shift location, and the grasp points need to be adjusted to minimize the risk of dropping the tray due to high load on the robot’s arms. Several works leverage wrench measurements to estimate geometric features, e.g. [4]–[7]. In our setup, we adopt the same fundamental force and torque relationship for quasi-static scenarios for the estimation of the COM.

Dynamic manipulation by pairs of robotics fingers achieves controlled manipulation of objects in an online fashion [8], [9]. However, our problem is concerned with the distribution of forces between grasp points, rather than fingers, and the nature of the constrained dual-arm manipulation task encourages a planning approach to the problem, as opposed to an online solution.

Crucially, in our situation, planning regrasp trajectories to adjust the grasp points on the object and improve balance [10]–[12] is not an option, because releasing the grasp

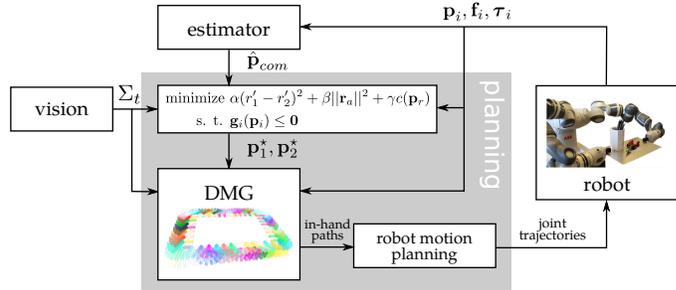


Fig. 1: A schematic overview of our proposed system. Wrench measurements are leveraged to produce an estimate,  $\hat{p}_{com}$ , which is used to initialize our planning pipeline. Visual information is employed to provide an accurate pose for the tray’s frame,  $\Sigma_t$ , in which the other quantities are expressed.

of one of the arms will lead to a stark increase in the wrenches experienced by the other. This significantly increases the chance of overcoming the maximum grasping force or exceeding the payload limitations of the single robot’s arm. Moreover, balancing by means of non prehensile grasps [13] as opposed to regrasping can lead to a substantial tilt of the tray, which is infeasible in a scenario in which the tray is carrying objects. Hence, we exploit dexterous manipulation to change the grippers’ contact points on the tray.

Dexterous manipulation includes the ability to move an object inside the hand without fully releasing the grasp, i.e., in-hand manipulation. Most in-hand manipulation works focus on a single hand system [14]–[16]. In contrast, we achieve *bimanual* dexterous manipulation, i.e., the dexterous manipulation task involves two distinct grasp points.

We consider robots equipped with parallel grippers, which are non-dexterous in nature. Such grippers can execute dexterous manipulation tasks by leveraging *extrinsic dexterity* [17]; extrinsic dexterity is the exploitation of supports external to the gripper, such as additional contact surfaces [18]–[20], gravity and friction [21], to compensate for the lack of degrees-of-freedom of the grippers. As external support, we exploit the redundancy of the dual-arm system to enable motions of the gripper along the tray without releasing the grasp: while one gripper moves, the other keeps a tight grasp on the tray so that the gripper’s fingers can slide along its surface without affecting its pose.

Our problem is related to the problem of attaining optimal grasp configurations through form or force closure, which is widely studied [22], and of maintaining a hold on the object under mass variations [16]. However, we focus on the distribution of the wrenches across arms, rather than on the stability of the grasps, which we assume can be

<sup>\*</sup>Diogo Almeida and Silvia Cruciani contributed equally to this work.

<sup>1</sup>Division of Robotics, Perception and Learning, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden {diogoalmeida, crucialiani, dani}@kth.se

<sup>2</sup>Dept. of Electrical Eng., Chalmers University of Technology, SE-412 96 Gothenburg, Sweden yiannis@chalmers.se

guaranteed by the robot as long as its payload limitations and gripping force limits are respected. Cooperative motion strategies for dual-arm robots have been developed, simplifying the command of a relative motion between robot arms, e.g., cooperative task space formulations [23], [24]. These approaches result in the simultaneous motion of the robot end-effectors, but this motion is undesirable in our case as the grasped tray could be dragged by the moving grippers. As such, we adopt a *discrete* motion of the robot end-effectors, i.e., planned joint trajectories are executed sequentially, one arm at a time.

The main contribution of this work is a complete system to balance wrenches on a dual-arm robot through a novel employment of bimanual dexterous manipulation, which is built on top of three contributions:

- 1) The use of bimanual sliding motions to change grasp configurations on an object.
- 2) The adaptation of Dexterous Manipulation Graphs (DMG) [18] to support bimanual manipulation.
- 3) The combined employment of vision and force-torque sensing, crucial to plan and execute the sliding trajectories.

## II. PROBLEM DESCRIPTION

In this section, we pose the wrench-balancing problem. We start by modeling the wrench distribution among the two robot end-effectors. Then, we relate the wrench at each grasp point to geometric features of the problem. Finally, we formally define the balancing task as the problem of optimizing a cost related to those quantities.

### A. Modeling

Consider a dual-arm robotic manipulator, where the two robot arms are jointly grasping a rigid tray. We are concerned with the distribution of wrenches across the two robot end-effectors: in general, the COM of the objects on the tray will be in a location which results in distinct wrenches at the two grasp points.

Let  $\Sigma_i \in SE(3)$  denote a reference frame, at the  $i$ -th grasp point,  $i \in \{1, 2\}$ .  $\Sigma_i$  is composed by a translational component,  $\mathbf{p}_i \in \mathbb{R}^3$  and an orientation, represented as a rotation matrix,  $\mathbf{R}_i \in SO(3)$ . We will assume that all quantities are expressed w.r.t a tray frame,  $\Sigma_t$ . The wrench at the grasp points can be written as  $\mathbf{w}_i = [\mathbf{f}_i^\top \quad \boldsymbol{\tau}_i^\top]^\top$ , where  $\mathbf{f}_i, \boldsymbol{\tau}_i \in \mathbb{R}^3$  denote, respectively, the forces and torques measured at the  $i$ -th point. Let the COM be defined as  $\mathbf{p}_{com}$  and consider the virtual sticks,  $\mathbf{r}_i$  which relate the grasp points with  $\mathbf{p}_{com}$  (see Fig. 2) and are defined as

$$\mathbf{r}_1 = \mathbf{p}_{com} - \mathbf{p}_1 \quad \mathbf{r}_2 = \mathbf{p}_2 - \mathbf{p}_{com}. \quad (1)$$

Additionally, we define the relative position of the gripping points as  $\mathbf{p}_r = \mathbf{p}_2 - \mathbf{p}_1$ . We can now express the forces at each grasp point as a function of the virtual sticks and the force  $\mathbf{f}_{com}$  exerted at the COM

$$\mathbf{f}_1 = (1 - r'_1)\mathbf{f}_{com} \quad \mathbf{f}_2 = (1 - r'_2)\mathbf{f}_{com}, \quad (2)$$

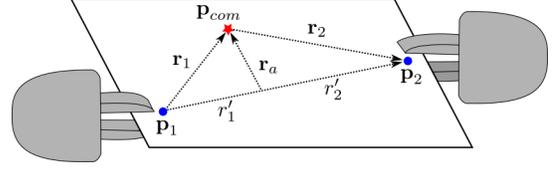


Fig. 2: Illustration of the relevant wrench balancing quantities. The two grippers grasp the tray at the grasp points  $\mathbf{p}_1$  and  $\mathbf{p}_2$ .  $\mathbf{p}_{com}$  is the projection of the COM on the tray's surface, where the force  $\mathbf{f}_{com}$  is applied.

where  $r'_i$  are the magnitude of the projections of the virtual sticks on the relative position, i.e.,

$$r'_i = \frac{\mathbf{r}_i^\top \mathbf{p}_r}{\|\mathbf{p}_r\|^2}. \quad (3)$$

Note that  $\mathbf{f}_{com} = \mathbf{f}_1 + \mathbf{f}_2$ . The torques at the contact areas are also related to the virtual sticks,

$$\begin{aligned} \boldsymbol{\tau}_1 &= \mathbf{r}_1 \times \mathbf{f}_{com} - \mathbf{p}_r \times \mathbf{f}_2 \\ \boldsymbol{\tau}_2 &= -\mathbf{r}_2 \times \mathbf{f}_{com} + \mathbf{p}_r \times \mathbf{f}_1, \end{aligned} \quad (4)$$

where we consider the effects that the dual-armed grasp have on each grasp point's measurements. Under the rigid grasp assumption, the torques at the two end-effectors are equal and rely on the distance between  $\mathbf{p}_r$  and  $\mathbf{p}_{com}$ ,

$$\mathbf{r}_a = \mathbf{p}_{com} - (r'_1\mathbf{p}_2 + r'_2\mathbf{p}_1), \quad (5)$$

such that the torques are given as a function of  $\mathbf{r}_a$ ,

$$\boldsymbol{\tau}_i = \mathbf{r}_a \times \mathbf{f}_{com}. \quad (6)$$

Note that the components of  $\mathbf{f}_{com}$  along the tray's surface normal will generate a torque about  $\mathbf{p}_r$ , which is generally undesirable, as the tray will fall if the robot does not compensate the torque. Minimizing  $\boldsymbol{\tau}_i$  reduces this risk.

**Definition 1** (Balanced Wrenches). *Let the tuple  $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_{com})$  denote a particular disposition of the robot grasp points and the gripped tray's center of mass. The wrenches measured at  $\Sigma_1, \Sigma_2$  are balanced when*

$$\mathbf{f}_e = \mathbf{f}_2 - \mathbf{f}_1 = \mathbf{0} \quad \text{and} \quad \boldsymbol{\tau}_e = \frac{\boldsymbol{\tau}_1 + \boldsymbol{\tau}_2}{2} = \mathbf{0}. \quad (7)$$

Thus, we can balance the experienced wrenches by finding a grasp configuration  $(\mathbf{p}_1^*, \mathbf{p}_2^*)$  which solves the optimization problem

$$\begin{aligned} \min_{\mathbf{p}_1, \mathbf{p}_2} \quad & \alpha \|\mathbf{f}_e\|^2 + \beta \|\boldsymbol{\tau}_e\|^2 \\ \text{s. t.} \quad & \mathbf{g}_1(\mathbf{p}_1) \leq \mathbf{0} \\ & \mathbf{g}_2(\mathbf{p}_2) \leq \mathbf{0}, \end{aligned} \quad (8)$$

where  $\alpha, \beta > 0$  are weights that compensate for the different units of forces and torques, and  $\mathbf{g}_i(\mathbf{p}_i)$  are constraints on the grasp point's locations. These are set such that we find solutions only on reachable areas of the tray.

To solve (8), we set the assumption that the force at  $\mathbf{p}_{com}$  will remain constant during the robot motion, i.e., objects will not slide or roll on the tray. In this case,  $\dot{\mathbf{f}}_{com} = \mathbf{0}$  and we can see that the dynamics of the errors defined in (7) are

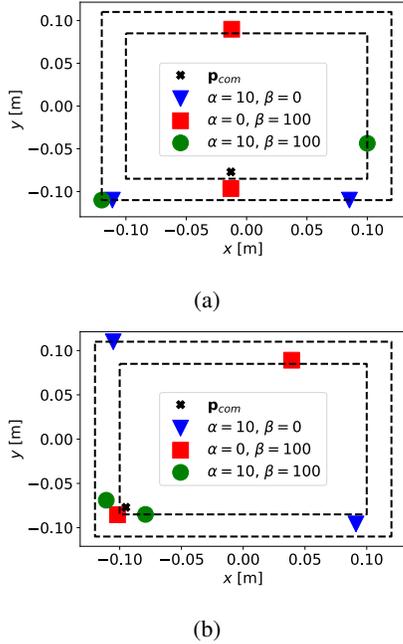


Fig. 3: Different balancing solutions can be obtained for the same  $\mathbf{p}_{com}$ , depending on the values of  $\alpha$  and  $\beta$ . Fig. 3a and 3b differ in the location of the COM. Note that the solution that minimizes both forces and torques in 3b is potentially unfeasible, due to the proximity of the points (green circles). The distance cost (11) can penalize these solutions.

completely determined geometrically by the locations of the grasp points and COM,

$$\mathbf{f}_e = (r'_1 - r'_2)\mathbf{f}_{com} \quad \text{and} \quad \boldsymbol{\tau}_e = \mathbf{r}_a \times \mathbf{f}_{com}. \quad (9)$$

Thus, we can pose an equivalent problem to (8) where the cost functional depends on  $r'_1$ ,  $r'_2$  and  $\mathbf{r}_a$ . Additionally, there are physical constraints in how close the fingertips of the robot manipulator can be. We address this by adding a nonlinear cost term which is a function of  $\mathbf{p}_r$ . We can now state the redefined optimization problem:

**Problem 1.** *To attain a balanced wrench configuration, we must place the system's grasp points at  $(\mathbf{p}_1^*, \mathbf{p}_2^*)$  which are the solution to the following optimization problem*

$$\begin{aligned} \min_{\mathbf{p}_1, \mathbf{p}_2} \quad & \alpha(r'_1 - r'_2)^2 + \beta\|\mathbf{r}_a\|^2 + \gamma c(\mathbf{p}_r) \\ \text{s. t.} \quad & \mathbf{g}_1(\mathbf{p}_1) \leq \mathbf{0} \\ & \mathbf{g}_2(\mathbf{p}_2) \leq \mathbf{0}, \end{aligned} \quad (10)$$

where  $\gamma > 0$  and the added cost

$$c(\mathbf{p}_r) = \begin{cases} 0 & \text{if } \|\mathbf{p}_r\|^2 > d_{min} \\ \|\mathbf{p}_r - d_{min}\mathbf{1}\|^2 & \text{otherwise,} \end{cases} \quad (11)$$

is a nonlinear term that penalizes magnitudes of  $\mathbf{p}_r$  smaller than  $d_{min} > 0$ , and  $\mathbf{1} \in \mathbb{R}^3$  is the vector whose entries are all 1.

In the absence of the constraints  $\mathbf{g}_i(\mathbf{p}_i)$ , all solutions where  $\mathbf{p}_{com}$  is centered between a line which crosses the two end-effectors will minimize both errors as defined in (7). This is a source of redundancy that can be exploited

by setting  $d_{min}$  in the cost function (11). However, given the constraints  $\mathbf{g}_i(\mathbf{p}_i)$ , centering  $\mathbf{p}_{com}$  as described above might not be feasible, which emphasizes the need to solve the optimization problem.

### B. Solving the balancing problem

Some insight on the possible solutions to (10) can be seen in Fig. 3a, where we optimize just for  $\mathbf{f}_e$  ( $\beta = 0$ ), just for  $\boldsymbol{\tau}_e$  ( $\alpha = 0$ ) or for both. It is clear that different values of the weights  $\alpha$  and  $\beta$  will lead to very different solutions. Furthermore, the location of  $\mathbf{p}_{com}$  may lead to solutions which are unfeasible, due to the close proximity of the grasp points (see Fig. 3b). These solutions are penalized by the distance cost (11).

Any solution to Problem 1 requires knowledge of  $\mathbf{p}_{com}$ . This is, in general, uncertain, as objects can be placed on the tray and shift the distribution of mass experienced by the robot. Once an estimate for  $\mathbf{p}_{com}$  is obtained, the tuple  $(\mathbf{p}_1^*, \mathbf{p}_2^*)$  is used to inform a DMG-based planning algorithm, which will produce suitable trajectories for the robot to regasp the tray.

## III. PROPOSED SYSTEM

In this section we describe the system we propose to solve the wrench balancing problem. We detail the overall architecture, and present a more comprehensive view of the estimation and motion planning components.

### A. Overview

Consider a dual-armed robot manipulator, equipped with a force-torque sensor at each wrist, that grasps a tray of known geometry. The tray is detected through an externally mounted camera. Once initialized, the available wrench measurements are utilized to estimate the location of the COM and subsequently used to solve Problem 1 with a nonlinear optimizer. The gripping configuration obtained from the optimizer is given to the DMG module, which in turn produces a path on the tray. This path is then interpolated and converted into joint trajectories. A joint trajectory controller is used to move the arms sequentially, thus assuring that the tray's motion is minimized during the execution. When the location of the COM changes, a user prompts the system to repeat the process. Fig. 1 presents a diagram that illustrates the system components and their interconnections.

### B. Estimation

To solve Problem 1, we employ estimates of the COM's location, denoted as  $\hat{\mathbf{p}}_{com}$ . We can update  $\hat{\mathbf{p}}_{com}$  from the measured wrenches, by leveraging equation (6) to construct an observation model for the estimator. We employ a Kalman filter with the process and observation models being given, respectively, by

$$\dot{\hat{\mathbf{p}}}_{com} = \mathbf{0} \quad \text{and} \quad \mathbf{z} = \mathbf{C}\hat{\mathbf{p}}_{com}, \quad (12)$$

where we assume that both models are affected by Gaussian white noise. The observation model's variables are given by

$$\mathbf{z} = \frac{\tau_1 + \tau_2}{2} - \mathbf{S}(\mathbf{f}_{com}) \left( \frac{\mathbf{f}_2^\top \mathbf{n}}{\mathbf{f}_{com}^\top \mathbf{n}} \mathbf{p}_2 + \frac{\mathbf{f}_1^\top \mathbf{n}}{\mathbf{f}_{com}^\top \mathbf{n}} \mathbf{p}_1 \right) \quad (13)$$

$$\mathbf{C} = -\mathbf{S}(\mathbf{f}_{com}),$$

where the matrix  $\mathbf{S}(\cdot)$  is skew-symmetric and realizes the cross product operation, i.e.,  $\mathbf{S}(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$ . The measurement vector  $\mathbf{z}$  in (13) relies on the assumption that the applied forces lay along the tray's surface normal  $\mathbf{n}$ . In this case, we can use the projections of the measured forces on the surface normal to get the values of  $r'_i$ , i.e.,

$$r'_1 = \frac{\mathbf{f}_2^\top \mathbf{n}}{\mathbf{f}_{com}^\top \mathbf{n}}, \quad \text{and} \quad r'_2 = \frac{\mathbf{f}_1^\top \mathbf{n}}{\mathbf{f}_{com}^\top \mathbf{n}}. \quad (14)$$

The estimator runs in an online fashion. However, once a manipulation task starts, we assume that the estimate available at the initial instant remains static until the task's completion.

### C. Planning

The planning step consists of two main components: an optimizer, to provide the target grasp points on the tray, and a Dexterous Manipulation Graph (DMG), to find solutions that move the grippers to the target grasp through in-hand manipulation. Finally, this in-hand path is used to generate a joint trajectory for the robot's arms.

1) *Optimizer*: Once an estimate  $\hat{\mathbf{p}}_{com}$  is obtained, we can solve (10). We use a sequential least squares programming (SLSQP) solver, where the inequality constraints correspond to the region within the tray where the robot gripping points can be located. These constraints can be, e.g., the two rectangular areas depicted in Fig. 3a-3b. In this step, we do not consider any other constraints that may exist due to the tray's geometry, such as protruding legs that can prevent the fingers' motion. Once a tuple  $(\mathbf{p}_1^*, \mathbf{p}_2^*)$  is obtained, we select a desired target angle for the gripper. This angle is redundant w.r.t the problem of balancing the wrenches, so we set it such that it maximizes the distance between the gripper and the edge of the tray. A path for the gripping points can now be produced by using a DMG constructed with the robot tray's model.

2) *Dexterous Manipulation Graph*: The DMG is a representation of how one finger of a single gripper can move along the object's surface. The process of building the DMG and exploiting it for planning in-hand manipulation is detailed in [18]. We provide a brief summary and explain how we adapt the DMG structure to in-hand manipulation problems that involve two grippers jointly carrying an object.

A DMG is a disconnected undirected graph, with disconnections depending on the constraints on translational and rotational motions of the fingertips at certain points, which are a function of the object's geometry. A DMG node is a tuple  $n = \langle \mathbf{p}_n, A_n \rangle$  containing information on the corresponding fingertip contact point  $\mathbf{p}_n$  and the possible range of orientations  $A_n$  between which the finger can continuously rotate when in contact at that point. An edge

$e_{nm}$  connects two DMG nodes  $n$  and  $m$  if the fingertip can slide between  $\mathbf{p}_n$  and  $\mathbf{p}_m$ , and  $A_n \cap A_m \neq \emptyset$ . A grasp is associated with two nodes, one node per finger, and the in-hand path planned to move to a different grasp configuration must take into account the feasibility of the motions for both fingers.

In our system, the object is held by two grippers at the same time, a scenario that does not fall into the problems solved with original DMG path planning. Therefore, we adapt the DMG planning to account for possible obstacles, including self-collisions. In this framework, the DMG provides two in-hand path solutions, one per gripper, each associated with its respective pair of DMG nodes.

For notational simplicity, in the following we associate one DMG node to one grasp; we remark that, while for simple tray shapes it can be assumed that the opposite node, corresponding to a second finger, will always be able to follow the first finger, for planning motions on generic shapes the opposite finger must be taken into account. The goal of the DMG planning step is to provide two paths  $\text{path}_1^*$  and  $\text{path}_2^*$  that move the grippers on the tray surface to the desired final grasps.

The two initial grasps on the tray correspond to the nodes  $n_1$  and  $n_2$  of the DMG. The target grasp positions  $\mathbf{p}_1^*$  and  $\mathbf{p}_2^*$  could be associated with more than one node, because of possibly discontinuous orientation ranges near the corners or other elements in the tray's shape. The choice of target orientation constrains the target grasps to two nodes  $n_1^*$  and  $n_2^*$ . Thanks to the DMG structure, the problem of planning in-hand motions becomes a graph search, from  $n_1$  to  $n_1^*$  and from  $n_2$  to  $n_2^*$ . However, the search for one node must consider that there is a second gripper holding a tray. To plan for one gripper, we consider that the other gripper, holding the tray, will be kept fixed. This discrete coordinated motion eases the planning problem and at the same time reduces the possibility of unpredicted motions of the tray by always keeping at least a firm grasp by one of the grippers on it.

We use Dijkstra's algorithm to find the path in the graph. However, we increase the cost of those edges that cross obstacles, the gripper and also objects placed on the tray, so that the resultant path is collision free. These collisions are checked by taking into account the whole body of the finger, and not just the fingertip contact point that slides on the tray. Which of the two grippers moves first depends of what order would result in the shortest in-hand path; in some cases (e.g. more obstacles on the tray, trays with additional elements such as feet) only one of the two sequences will provide a valid solution. Moreover, attaining a balanced wrench distribution does not require a particular gripper to reach a particular desired point. As such, we exploit the DMG solutions to find the shortest combined in-hand path that will result in the two grippers holding the tray at  $\mathbf{p}_1^*$  and  $\mathbf{p}_2^*$ . This procedure is explained in Algorithm 1, in which we have omitted the cases of null paths in some steps for the sake of simplicity. The overall output contains the in-hand path for each gripper, and also the order in which the grippers have to be moved.

---

**Algorithm 1** Find In-Hand Paths

---

**Input:** DMG,  $n_1, n_2, n_1^*, n_2^*$ **Output:**  $\text{path}_1^*, \text{path}_2^*, \text{order}$ 

```
procedure PATHSDMG( $n_{start}, n_{goal}, m_{start}, m_{goal}$ )
  DMG.ADDGRIPPEROBSTACLE( $m_{start}$ )
   $\text{path}_n \leftarrow$  DMG.SHORTESTPATH( $n_{start}, n_{goal}$ )
  DMG.REMOVEGRIPPEROBSTACLE()
  DMG.ADDGRIPPEROBSTACLE( $n_{goal}$ )
   $\text{path}_m \leftarrow$  DMG.SHORTESTPATH( $m_{start}, m_{goal}$ )
  DMG.REMOVEGRIPPEROBSTACLE()
  return  $\text{path}_n, \text{path}_m$ 
end procedure
procedure ORDEREDPATHS( $n_1^s, n_2^s, n_1^g, n_2^g$ )
   $\text{path}_1^1, \text{path}_2^1 \leftarrow$  PATHSDMG( $n_1^s, n_1^g, n_2^s, n_2^g$ )
   $\text{order}^1 \leftarrow 1, 2$ 
   $\text{path}_2^2, \text{path}_1^2 \leftarrow$  PATHSDMG( $n_2^s, n_2^g, n_1^s, n_1^g$ )
   $\text{order}^2 \leftarrow 2, 1$ 
  if  $\text{path}_1^1.\text{SIZE} + \text{path}_2^1.\text{SIZE} < \text{path}_1^2.\text{SIZE} + \text{path}_2^2.\text{SIZE}$ 
    return  $\text{path}_1^1, \text{path}_2^1, \text{order}^1$ 
  else
    return  $\text{path}_1^2, \text{path}_2^2, \text{order}^2$ 
end procedure
 $\text{path}_a, \text{path}_b, \text{order}_{ab} \leftarrow$  ORDEREDPATHS( $n_1, n_2, n_1^*, n_2^*$ )
 $\text{path}_c, \text{path}_d, \text{order}_{cd} \leftarrow$  ORDEREDPATHS( $n_1, n_2, n_2^*, n_1^*$ )
if  $\text{path}_a.\text{SIZE} + \text{path}_b.\text{SIZE} < \text{path}_c.\text{SIZE} + \text{path}_d.\text{SIZE}$ 
   $\text{path}_1^*, \text{path}_2^*, \text{order} \leftarrow \text{path}_a, \text{path}_b, \text{order}_{ab}$ 
else
   $\text{path}_1^*, \text{path}_2^*, \text{order} \leftarrow \text{path}_c, \text{path}_d, \text{order}_{cd}$ 
```

---

3) *Robot's motion*: A DMG solution is composed by two paths, one for each gripper's fingertips on the tray's surface. These paths are interpolated into a finer set of points which obey the contact constraints with the tray. Then, the corresponding joint positions are obtained with TRAC-IK [25], which ensures compliance with the kinematic constraints of the robot. These positions are converted into suitable joint trajectories for the two arms through trapezoidal integration. We execute the trajectories sequentially, one arm at a time, while the static arm grasps the tray rigidly to prevent undesired motions due to the friction forces of the sliding finger.

## IV. EXPERIMENTS

### A. Experimental setup

We implemented our system on an ABB Yumi robot, which is a good example of a collaborative robot with very limited payload. As such, regrasping the tray is not an option, as the wrench on a single arm can easily exceed its payload limits or gripping force. Our robot is equipped with two optoforce HEX 6-axis force torque sensors. We obtain the tray's pose by using AprilTag fiducial markers [26], detected by an externally mounted Microsoft Kinect2 sensor. The tray used in the experiments consists of a rectangular aluminum surface, with an area of  $300 \times 200$  mm and which weighs 186 grams. Note that the combined weight of the tray, gripper and force torque sensor (approximately 600 grams) is enough to overcome Yumi's single arm payload limits (500 grams), which emphasizes the need for a bimanual grasp.

### B. Results

We report two types of experiments, where we successively run our system to balance either forces (i.e.,  $\beta = 0$ ) or torques (i.e.,  $\alpha = 0$ ). In all experiments we set  $\gamma = 1000$ , with  $d_{\min} = 0.15$ m. The COM estimator was run with  $0.1\mathbf{I}$  as the assumed observation covariance noise and  $0.01\mathbf{I}$  for the process noise. The successful force balancing experiments relied on solutions that minimized both forces and torques. Given the constraints on our real system, reflected on the constraints of the optimization problem, solutions for  $\alpha, \beta \neq 0$  are very similar to the successful force balancing solutions and as such we omit them for conciseness.

1) *Force balancing*: We consider balancing just the forces at the two gripping points, i.e., achieving  $\mathbf{f}_e = \mathbf{0}$ . Note that without considering torques, achieving this goal can result in a less stable grasp on the tray, since a configuration that balances forces perfectly might be far from  $\mathbf{p}_r$ , as seen in Figs. 3a-3b. Setting  $\beta = 0$  in eq. (10) and ignoring the torques in the optimization problem can thus lead to gripping solutions which create a large lever effect. To circumvent this problem, we restrict the gripping points to remain on their respective initial sides of the tray by adjusting the constraints  $\mathbf{g}_i(\mathbf{p}_i)$ . The successful force balancing experiments do not increase the torque error, but actually contribute to lowering it, Fig. 4a. We illustrate the results for consecutive executions with different initial conditions in Fig. 4b. In all experiments,  $\alpha$  was set to 10. The average execution time was 45s, with a minimum of 38s and a maximum of 53s. Note that all reported executions have reduced the torque error, from an average initial error of 0.15 N·m to an average final error of 0.025 N·m.

2) *Torque balancing*: In these set of experiments, the torque weight in (10) is set to  $\beta = 100$ , with  $\alpha = 0$ . Our results with the force balancing experiments suggest that it is indeed crucial to minimize the torques, as these are a contributing factor to an unstable equilibrium of the tray. We depict the evolution of the force and torque errors for 20 runs in Fig. 4c. The average execution time was 20s, with a minimum of 15s and a maximum of 27s. The faster execution times w.r.t the force balancing results stem from shorter computed paths. Despite an average increase in the force error from 1 to 2 N, the resulting grasps were stable. A similar conclusion could not be made for the force balancing problem, where only the trajectories that happened to minimize torques were successful.

### C. Discussion

Bimanual dexterous regrasping is a suitable approach to the wrench balancing problem, where releasing one of the grasps on the tray can lead to a catastrophic system failure due to excessive wrenches. Our system is able to execute the manipulation task successfully by reducing the experienced wrench imbalance. During the execution of our experiments, we observed the critical importance of balancing the torques. In fact, when only balancing forces, successful executions consist only of trajectories which contributed in reducing torques. Trajectories which balanced forces but increased the

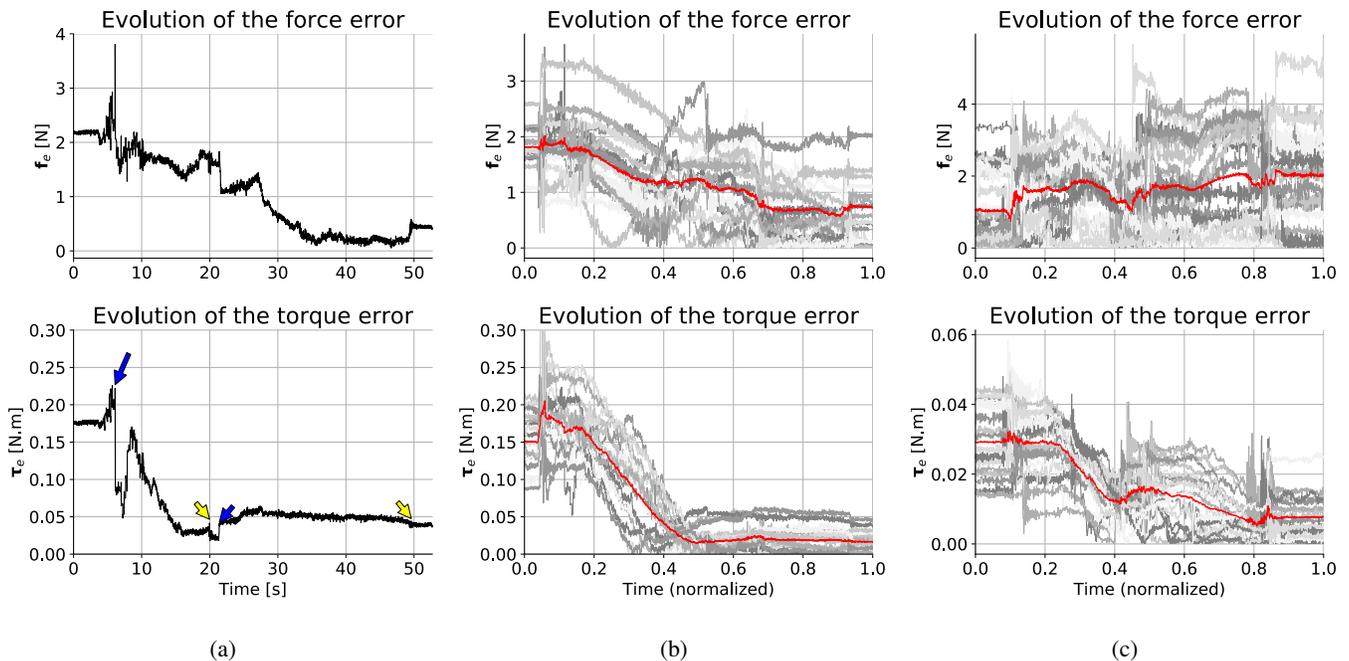


Fig. 4: Experimental results of our force balancing system. We depict the outcome of a single execution, where the system was tasked with **minimizing forces**, in Fig. 4a. The blue and yellow arrows denote, respectively, the moments when a gripper starts and concludes a sliding trajectory. **15 consecutive executions** of the system for the same goal are portrayed in Fig. 4b, with the average errors plotted in red. The outcome of **20 consecutive executions** of our system when tasked with **minimizing torques**, are plotted in Fig. 4c.

torques resulted in system failure due to exceeding payload limitations, even for lightweight objects. On the contrary, we succeeded in balancing torques even for trajectories where the relative force increased. Some trajectories depicted in Fig. 4 are affected by small pivoting and tilting motions of the tray w.r.t Yumi’s fingertips. The adoption of fingertips with a larger contact surface would be a straightforward addition to our system and would guarantee a firm grasp at all times.

We rely on the availability of IK solutions for the desired paths on the tray, and for the area where the gripping points move to be clear of dynamic obstacles. We can address these issues by exploiting the inherent redundancy of this problem and incorporating online object avoidance in future work.

*1) Redundancy exploitation:* There are two main sources of redundancy on our problem that can be explored to increase the range of possible solutions. These consist in the selection of target angles of the fingers at the goal configuration, and in planning an absolute motion for the tray while both grasps on it are tight.

The wrenches experienced by both arms rely only on the location of the gripping points and, when commanding the end-effectors’ motion, the fingers’ orientation is redundant w.r.t the balancing problem. In this work, this was exploited simply by setting orientations which are normal to the tray’s edge, minimizing the chances of collision with the tray during motion. However, the finger’s orientation can be planned, together with an absolute motion on the tray, to find solutions which, e.g., avoid joint limits. In our current setup, if there are no IK solutions for a desired trajectory on the tray, we do not execute the balancing task.

*2) Incorporating object avoidance:* We have used vision to detect the tray’s pose on the robot base frame. We can, however, employ vision to detect potentially obstructed paths on the tray due to, e.g., the location of placed objects. Our new version of DMG can plan paths taking such obstacles into account. Furthermore, the combined employment of wrench and vision measurements allows the two sensing means to complement each other: wrench measurements relate to dynamical quantities that cannot be observed through vision. In contrast, vision provides geometric information that is not observable from wrench measurements alone.

## V. CONCLUSIONS

In this paper, we presented a strategy for balancing the wrench distribution across two robot arms that are jointly grasping a rigid tray. This task is executed through discrete bimanual manipulation, where two support points remain on the tray at all times. As objects are placed on the tray, its COM changes location, which results in an uneven force distribution across the arms. We showed how the problem of balancing these two components can be solved geometrically, and how to employ an optimization algorithm to find desired grasp locations on the tray which will result in a better wrench distribution. Furthermore, the measured wrenches are leveraged to estimate the location of the COM. As releasing the tray to regrasp it will often lead to a large increase in the wrench at one of the robot arms, we execute sliding trajectories, computed by DMG. To the best of our knowledge this is the first reported experimental execution of bimanual dexterous manipulation.

## REFERENCES

- [1] M. Uchiyama, N. Iwasawa, and K. Hakomori, "Hybrid position/Force control for coordination of a two-arm robot," in *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, vol. 4, March 1987, pp. 1242–1247.
- [2] F. Caccavale, P. Chiacchio, and S. Chiaverini, "Task-space regulation of cooperative manipulators," *Automatica*, vol. 36, no. 6, pp. 879 – 887, 2000.
- [3] L. Yan, Z. Mu, W. Xu, and B. Yang, "Coordinated compliance control of dual-arm robot for payload manipulation: Master-slave and shared force control," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 2697–2702.
- [4] D. Verscheure, I. Sharf, H. Bruyninckx, J. Swevers, and J. De Schutter, "Identification of contact dynamics parameters for stiff robotic payloads," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 240–252, April 2009.
- [5] Y. Karayiannidis, C. Smith, F. E. Viña, and D. Kragic, "Online contact point estimation for uncalibrated tool use," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 2488–2494.
- [6] D. Almeida, F. E. Viña, and Y. Karayiannidis, "Bimanual folding assembly: Switched control and contact point estimation," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, Nov 2016, pp. 210–216.
- [7] D. Almeida and Y. Karayiannidis, "Cooperative Manipulation and Identification of a 2-DOF Articulated Object by a Dual-Arm Robot," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1–5.
- [8] S. Arimoto, P. T. A. Nguyen, H.-Y. Han, and Z. Doulgeri, "Dynamics and control of a set of dual fingers with soft tips," *Robotica*, vol. 18, no. 1, p. 71–80, 2000.
- [9] Z. Doulgeri, J. Fasoulas, and S. Arimoto, "Feedback control for object manipulation by a pair of soft tip fingers," *Robotica*, vol. 20, no. 1, p. 1–11, 2002.
- [10] Z. Xian, P. Lertkultanon, and Q. C. Pham, "Closed-chain manipulation of large objects by multi-arm robotic systems," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 1832–1839, Oct 2017.
- [11] P. Lertkultanon and Q. Pham, "A certified-complete bimanual manipulation planner," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1355–1368, July 2018.
- [12] L. Chen, L. F. C. Figueredo, and M. Dogar, "Manipulation planning under changing external forces," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 3503–3510.
- [13] R. Krug, Y. Bekiroglu, D. Kragic, and M. A. Roa, "Evaluating the quality of non-prehensile balancing grasps," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1–6.
- [14] N. C. Dafle, R. Holladay, and A. Rodriguez, "In-hand manipulation via motion cones," in *Proceedings of Robotics: Science and Systems*, June 2018.
- [15] OpenAI, M. Andrychowicz, B. Baker, M. Chociej, R. Józefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, "Learning dexterous in-hand manipulation," *CoRR*, 2018. [Online]. Available: <http://arxiv.org/abs/1808.00177>
- [16] K. Hang, M. Li, J. A. Stork, Y. Bekiroglu, F. T. Pokorny, A. Billard, and D. Kragic, "Hierarchical fingertip space: A unified framework for grasp planning and in-hand grasp adaptation," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 960–972, Aug 2016.
- [17] N. C. Dafle, A. Rodriguez, R. Paolini, B. Tang, S. S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1578–1585.
- [18] S. Cruciani, C. Smith, D. Kragic, and K. Hang, "Dexterous Manipulation Graphs," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2040–2047.
- [19] N. Chavan-Dafle and A. Rodriguez, "Prehensile pushing: In-hand manipulation with push-primitives," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, Sept 2015, pp. 6215–6222.
- [20] D. Almeida and Y. Karayiannidis, "Dexterous manipulation with compliant grasps and external contacts," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 1913–1920.
- [21] F. E. Viña B., Y. Karayiannidis, K. Pauwels, C. Smith, and D. Kragic, "In-hand manipulation using gravity and controlled slip," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, pp. 5636–5641.
- [22] A. Bicchi, "On the closure properties of robotic grasping," *The International Journal of Robotics Research*, vol. 14, no. 4, pp. 319–334, 1995.
- [23] P. Chiacchio, S. Chiaverini, and B. Siciliano, "Direct and inverse kinematics for coordinated motion tasks of a two-manipulator system," *Journal of dynamic systems, measurement, and control*, vol. 118, no. 4, pp. 691–697, 1996.
- [24] H. A. Park and C. S. G. Lee, "Extended Cooperative Task Space for manipulation tasks of humanoid robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 6088–6093.
- [25] P. Beeson and B. Ames, "TRAC-IK: An open-source library for improved solving of generic inverse kinematics," in *Proceedings of the IEEE RAS Humanoids Conference*, Seoul, Korea, November 2015.
- [26] E. Olson, "AprilTag: A robust and flexible visual fiducial system," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2011, pp. 3400–3407.