Proceedings of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Dual Arm Manipulation using Constraint Based Programming ⋆

Yuquan Wang, Francisco Viña, Yiannis Karayiannidis, Christian Smith, Petter Ögren

All authors are with the Center for Autonomous Systems (CAS), at the Royal Institute of Technology (KTH), SE-100 44, Stockholm, Sweden. Email of corresponding author: `yuquan at kth.se`.

Abstract:
In this paper, we present a technique for online generation of dual arm trajectories using constraint based programming based on bound margins. Using this formulation, we take both equality and inequality constraints into account, in a way that incorporates both feedback and feedforward terms, enabling e.g. tracking of timed trajectories in a new way. The technique is applied to a dual arm manipulator performing a bi-manual task. We present experimental validation of the approach, including comparisons between simulations and real experiments of a complex bimanual tracking task. We also show how to add force feedback to the framework, to account for modeling errors in the systems. We compare the results with and without feedback, and show how the resulting trajectory is modified to achieve the prescribed interaction forces.

1. INTRODUCTION

For robotics to move from the factory floors to unstructured domestic environments, progress is needed in several areas of robotic technology. One such area is dual arm manipulation, where the human-like structure of a robot, such as the one in Figure 1, is exploited to perform tasks in environments originally intended for humans, as explained by Smith et al. [2012].

The potential benefits of endowing robots with dual arms fall into four main categories. First, using tools and workflows designed for humans is easier if the kinematic structure of the robot is similar to a human according to Kemp et al. [2007], Fuchs et al. [2009], Bloss [2010], Kr uger et al. [2011]. Second, teleoperation is easier if the robot is similar to the operator [Jau, 1988, Yoon et al., 1999, Kron and Schmidt, 2004, Buss et al., 2006, Taylor and Seward, 2010]. Third, the use of the two arms can either provide additional strength and precision by cooperating as a parallel manipulator, or provide flexibility and speed by doing two separate tasks simultaneously as discussed by Lee and Kim [1991]. Fourth, the two arms are able to perform task that are inherently bi-manual as reported by Chiacchio and Chiaverini [1998], Caccavale et al. [2000], i.e., tasks that require motion of both arms to be carried out efficiently.

In this paper we put the focus on such bi-manual tasks, which often include significant redundancies, a fact that makes them well suited to constraint based programming approaches. Examples include the following:

- Handing over a mug with one end-effector to a human and picking up a box of tea with another.

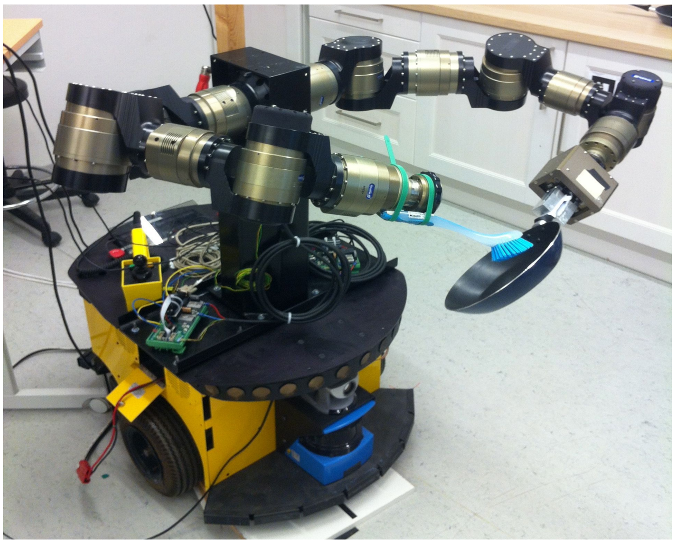

Fig. 1. The dual arm robot performing the task of cleaning a frying pan.

- While a human is lifting one side of a table the robot manipulates the other side with two grippers.
- Holding a workpiece with one hand and using some related tool with the other hand, e.g. manually cleaning a small object.

While performing bi-manual tasks, the robot need to simultaneously take the following secondary constraints into consideration: avoid internal collisions, avoid external collisions, avoid singularities, and finally keep the robot arms and the manipulated objects in the camera field of view.

Constraint based programming for robot motion generation has received a lot of attention, as it enables the execution of highly complex robot tasks. Constraint based

⋆ This work has been supported by the Swedish Research Council (VR), the Swedish Foundation for Strategic Research (SSF), and the European Union FP7 project RoboHow.Cog (FP7-ICT-288533).

programming work can be found in the early publications by Samson et al. [1991], Seraji [1989], Peng and Adachi [1993], while recently the framework iTaSC (Instantaneous Task Specification using Constraints) by De Schutter et al. [2007], Decré et al. [2009], Smits et al. [2009] has built upon it. The strength of constraint based programming is that it facilitates the formulation and solution of a wide range of robot control problems, where a number of different, possibly contradicting constraints, or objectives, needs to be taken into account. In this paper, we will present a new variation on constraint based programming, and apply it to do online dual arm manipulation.

The main contribution of the present work is the theoretical extension of margin based constraint based programming using inequality constraints, to also include time dependent equality constraints in a compact and uniform way. In order to demonstrate the applicability of the proposed approach to a dual arm problem, we take a bi-manual dish washing task as a proof of concept example.

We model the dish washing task with specified contact force with a set of time dependent equality constraints and one more inequality constraint. The secondary constraints are specified with another set of inequality constraints. We treat these constraints with the proposed method and the result is verified both in simulation and on a physical robot platform.

The outline of the paper is as follows. In Section 2 we relate the proposed approach to the state of the art. In Section 3 we describe the proposed version of constraint based programming and contrast it to the state of the art. Section 4 then formalizes the dual arm manipulation problem. The proposed solution is presented in Section 5. The simulation validation is followed by additional experiments on a real robot in Section 6. Finally, we conclude the paper in Section 7.

2. RELATED WORK

In general, the approaches to generating motion of redundant manipulators can be divided into two categories, global (offline) and local (online). In the offline approaches, we plan joint space trajectories such that we meet the desired objectives while fulfilling other constraints. As reported by Patel et al. [2005], these methods are computationally expensive and often require the kind of structured environments that can be found in factories, but also generate very efficient solutions, e.g., by solving minimum time problems.

Online approaches on the other hand, need to be less computationally expensive to meet real time requirements. They can handle unstructured environments with moving obstacles, such as domestic environments with humans nearby, but might produce less efficient solutions and might even fail to solve problems in really difficult cases, such as maze-like environments.

These complementary qualities may be best exploited by combining algorithms of both types. An offline algorithm may generate high-level plans and trajectories, that are then carried out by an online algorithm that performs the desired set of subtasks while at the same time ensuring that the secondary constraints are satisfied. In this paper, we put the focus on the online (local) approach. We propose a controller that can perform the assigned tasks in an efficient manner while satisfying the desired constraints. Note that the *generation* of such assigned tasks or constraints is not treated in the current work, one could specify task constraints, e.g. using the *iTask* approach described by De Schutter et al. [2007], Smits et al. [2009].

The proposed approach uses constraint based programming, which has its roots in the concepts of the *additional tasks* by Seraji [1989], the *user defined objective functions* by Peng and Adachi [1993], and the *sub-tasks* by Tatlicioglu et al. [2008]. Similar ideas were used in the *Stack of Tasks* approach by Mansard and Chaumette [2007], Mansard et al. [2009], the *iTask* approach by De Schutter et al. [2007], Smits et al. [2009], and in a variation using Quadratic programming that was proposed by Zhang et al. [2004], Zhang and Mai [2007].

However, as the approaches listed above assume that the main objective is defined with respect to the desired, possibly time varying, position and orientation of the end effector, they address all the secondary tasks using the so-called self motion, in the orthogonal space of the end-effector Jacobian.

By introducing scalar inequalities and bound margins as reported by Ögren [2008], Ögren and Robinson [2011], Ögren et al. [2012], our work can address additional tasks that are not limited to the null space of the Jacobian, and is also not limited to a number of additional constraints equal to the degree of redundancy, since inequality constraints do not reduce the dimensionality of the feasible set in joint space. The concept of using inequalities in this type of constraint based programming was earlier applied to dual arm manipulation by Ögren et al. [2012], mobile robot obstacle avoidance by Ögren [2008] and surveillance UAV control by Ögren and Robinson [2011].

Much of the work in this paper, and in Ögren [2008], Ögren and Robinson [2011], Ögren et al. [2012] is related to the strong contributions reported in Kanoun et al. [2009], Kanoun [2012]. It should however be noted that the work has been developed independently, as for example one of the major topics of Kanoun [2012], adding inequalities, was described in Ögren [2008]. This work however, goes beyond Kanoun [2012] in that we add exact tracking of timed trajectories, force feedback constraints, as well as experimental validation of the whole framework. Furthermore, different from using the *active set method* by Kanoun [2012] or iterating through inequality constraint one by one such as Kanoun et al. [2009], we solve one QP once for all. In order to ensure the solvability of this QP and prioritize different constraints, we weight the slack variables which are assigned to each constraint and minimize their weighted sum in the objective.

3. A NEW VARIATION ON CONSTRAINT BASED PROGRAMMING

In this section we will describe the proposed version of constraint based programming using bound margins. The inequalities part of the proposed approach was described