Multi-sensory Feedback Control in Door Approaching and Opening

Tomasz Winiarski, Konrad Banachowicz, and Dawid Seredyński

Warsaw University of Technology, Warsaw, Poland
tmwiniarski@gmail.com
http://robotics.ia.pw.edu.pl

Abstract. In the article the robotic system behavior is investigated for the complex door opening task. The system consists of the 7-DOF KUKA LWR4+ manipulator, which is controlled in an impedance way and the BarrettHand gripper, which is controlled in a position way. The system utilizes multi-sensory feedback. The visual feedback is used to roughly localize door and to plan a door approach trajectory. The tactile feedback detects the contact with the door, and handle and determines an exact contact position with the handle. The system does not form a grip in a door opening stage, but the contact between the robot and the door is maintained by the gripper's fingers (with intrinsic backlash), which are pushing the handle from its one side. This concept allows to open the door when there are obstacles in the neighborhood of the handle (e.g. door jamb or frame), which make the grip impossible.

Keywords: service robot, impedance control, tactile sensing.

1 Introduction

Demographic and civilization changes are encouraging robotic community to constantly research on service robots. A number of research works regarding robotic door opening confirm, that this task will be vital for the future of service robots, which will operate in the human oriented environment. To open a door successfully, several stages have to be completed. At first, a door and its components are localized on the base of information from visual subsystem [1] to plan manipulator's end–effector approach to reach the door handle and get in contact with it [2]. When the end–effector is approaching stiff objects (e.g. door handle or Rubik's Cube [3, 4]) the impact resistance is essential. This problem can be partially solved in the systems with indirect force control by applying parallel visual – force control [5]. In our article we investigate similar cooperation of visual subsystem with robot controller.

The door opening itself can be realized in various ways. For stiff connection of the manipulator's end–effector and door, there is no need to estimate door kinematics [6]. The most of the methods base on the door kinematic estimation, because for common grippers and door handles the stiff junction can not be guaranteed. The research was conducted on both cases: the velocity-controlled

D. Filev et al. (eds.), *Intelligent Systems'2014*,
Advances in Intelligent Systems and Computing 323, DOI: 10.1007/978-3-319-11310-4_6

manipulator with force sensing capabilities at the end-effector [7] and impedance controlled manipulator with torque controllers in its joints [8, 9]. The method of kinematic modeling of unknown 3D articulated objects was studied in the interactive robotic system [10]. In the newest papers [11] the general approach is proposed to two similar cases: a door opening and a drawer opening.

The robots, which are controlled to open a door, use the force or tactile feedback to detect and maintain contact, while the motion is in progress. In the work [12] the robot is indirectly controlled using the readings of custom force sensors located in the gripper phalanges, while typically similar systems acquire general force readings from a six-axis sensor mounted in the manipulator's wrist [13]. The robot presented in [14] is controlled in a similar way, but the force readings are aggregated from both tactile sensors in phalanges and force-torque sensor mounted in the wrist. In general the algorithms are planning and executing the motion, where the door knob or handle is grabbed from both sides to maintain the grip, while the door is moving.

In our work we specify and build the system (sec. 2), where the 7-DOF KUKA LWR4+ manipulator is controlled in impedance way (sec. 2.1) and the Barrett-Hand gripper is controlled in position way with tactile feedback (sec. 2.2). The simple visual subsystem (sec. 2.3) uses markers to roughly localize the door, then the controller plans the end–effector approach trajectory and finally robot gets in contact with door handle. As the contribution we investigate the initial manipulator approach (sec. 3.1) and subsequent door opening (sec. 3.2) with usage of information from tactile sensors of the three finger gripper. The control algorithm was especially developed to perform the task for the assumed concept of contact. The point is that the system do not form the grip in door opening stage, but the contact between the robot and the a door is maintained by the gripper fingers (with intrinsic backlash), which are pushing the handle from its one side. The fingers of typical grippers are quite thick, so this concept allows to open the door when obstacles (e.g. door jamb or frame or other handles), which are close to the handle, make the grip impossible. Several experiments (sec. 4) have been performed to verify the solution. Finally, the conclusions (sec. 5) summarize our work.

2 Control System

The general system structure is based on embodied agent approach to system development [15] and is depicted in fig. 1. Control Subsystem c with task specific door approaching and opening algorithm (sec. 3) is implemented in ROS [16] open-source, meta-operating system. The usage of ROS is adequate for fast prototyping of control algorithms and further experimental results analysis of systems equipped with grippers [17]. The Control Subsystems communicates with two Virtual Effectors e implemented in real time Orocos [18] robot programming framework through communication buffers b. The Virtual Effector e_m aims to control the Real Effector E_m of KUKA LWR4+ manipulator (sec. 2.1). The role of the Virtual Effector e_g is to control the Real Effector E_g of namely the

BarrettHand gripper (sec. 2.2). In general, Virtual Effectors were implemented to constitute universal interface or Hardware Abstraction Layer (HAL) between Control Subsystem and Real Effectors (Hardware). The aggregated data from visual subsystem (sec. 2.3) is transmitted from the Virtual Receptor r_v to the Control Subsystem c. The images originate from camera device, labeled as Real Effector R_v.

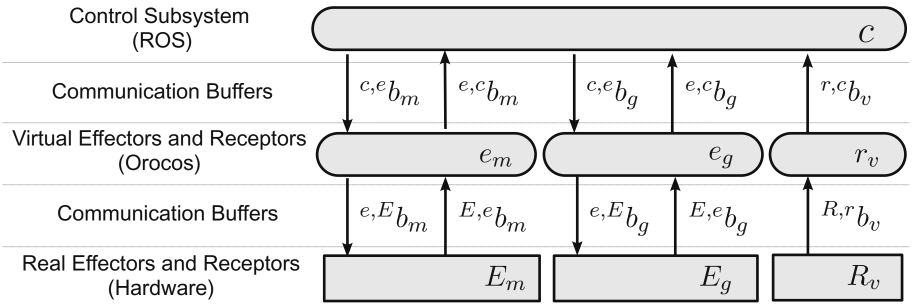


Fig. 1. General system structure

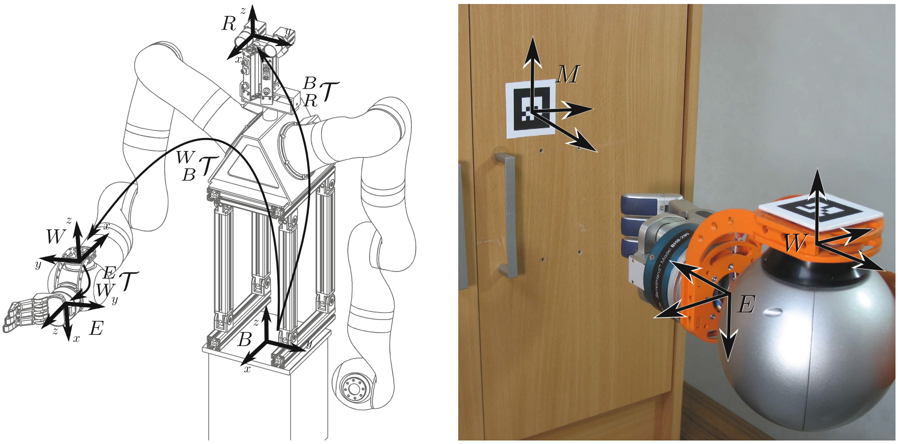
Five reference frames, which are used by the algorithm, are located in the experimental setup depicted in fig. 2:

- $\boldsymbol{B}$ – base frame of the robot,
- $\boldsymbol{R}$ – camera frame,
- $\boldsymbol{W}$ – frame of the last, seventh link of the manipulator (the wrist),
- $\boldsymbol{E}$ – gripper (end–effector) frame,
- $\boldsymbol{F}$ – finger's distal link frame,
- $\boldsymbol{C}$ – frame for contact point between handle and finger,
- $\boldsymbol{M}$ – door marker frame.

2.1 Control of Manipulator

The Control Subsystem communicates with the Virtual Effector of the manipulator using two communication buffers. The communication buffer ${}^{c,e}b_m$ consists of 6×6 diagonal Cartesian stiffness matrix $\boldsymbol{K}_t$ specified by the 6×1 stiffness vector $\boldsymbol{k}$ laying on the diagonal of this matrix, 6×1 normalized damping ratio vector $\boldsymbol{\xi}$, desired equilibrium point $\boldsymbol{r}_t$ (consisting of 3×1 position vector supplemented by a unit quaterion) and trajectory segment time t. The communication buffer ${}^{e,c}b_m$ contains a vector $\boldsymbol{q}(7 \times 1)$ of measured joint positions.

The manipulator is controlled using the Cartesian impedance [19]. The impedance control law creates the virtual spatial spring displacement $\Delta\boldsymbol{r}$ [20] between he pose of the wrist $\boldsymbol{W}$ measured in relation to the base $\boldsymbol{B}$ and represented by 6×1 vector $\boldsymbol{r}_w$ (consisting of the position vector supplemented by a unit



(a) (b)

Fig. 2. The experimental setup (a) robot arm and head with frames $\boldsymbol{B}$, $\boldsymbol{R}$, $\boldsymbol{W}$ and $\boldsymbol{E}$. (b) manipulator's end–effector approaching the door

quaternion) and the commanded equilibrium pose $\boldsymbol{r}_d$ represented in the same way. To ensure a stability of the system a damper is also included. The 6×1 desired force $\boldsymbol{F}_d$ at wrist frame is calculated (1) as a superposition of forces originating from the damper and the virtual spring:

$$\boldsymbol{F}_d = \boldsymbol{K}_c \Delta \boldsymbol{r} - \boldsymbol{D}_c(\boldsymbol{q}, \boldsymbol{\xi})\dot{\boldsymbol{r}}_w, \quad (1)$$

$$\boldsymbol{\tau} = \boldsymbol{J}^T(\boldsymbol{q})\boldsymbol{F}_d \ . \quad (2)$$

In general, the pose $\boldsymbol{r}_w$ in terms of $\boldsymbol{q}$ is computed using direct kinematics. Trajectory generation provides target equilibrium points $\boldsymbol{r}_d$ for the impedance controller by interpolating between the trajectory points $\boldsymbol{r}_t$ (using linear interpolation for position and spherical interpolation for rotation). The commanded stiffness $\boldsymbol{K}_c$ is interpolated in the analogous way basing on $\boldsymbol{K}_t$. The damping term is composed of a configuration dependent 6×6 damping matrix $\boldsymbol{D}_c$ and the velocity of the wrist frame $\dot{\boldsymbol{r}}_w$. The damping matrix is calculated at every control cycle using the double-diagonalization [21] method and is parametrized by the vector $\boldsymbol{\xi}$ representing normalised damping along the main directions of the wrist frame.

Following this, the force vector is transformed from wrist frame into joint torques by the 6×7 transposed Jacobian $\boldsymbol{J}^T$ of the manipulator. The commanded 7×1 torque vector $\boldsymbol{\tau}$ is then transfered to Real Effector using ${}^{e,E}b_m$ communication buffer. In reply Real Effector transfers measured position of joints $\boldsymbol{q}$ using ${}^{E,e}b_m$ communication buffer.

2.2 Control of Gripper

Control Subsystem communicates with the Virtual Effector of gripper using two communication buffers. The communication buffer ${}^{c,e}b_g$ consists of 4×1 desired joint position vector $\boldsymbol{q}_d$ and 4×1 maximum joint velocity vector $\boldsymbol{\omega}$. The communication buffer ${}^{e,c}b_g$ consists of 8×1 measured joint position vector $\boldsymbol{q}_m$, and 4×24 measured tactile force matrix $\boldsymbol{f}$. The difference in dimension of $\boldsymbol{q}_d$ and $\boldsymbol{q}_m$ results from the BarrettHand gripper construction. It has 4 motors propelling 8 joints, which are coupled in pairs. The Virtual Effector communicates with Real Effector with other two buffers. The communication buffer ${}^{e,E}b_g$ consists of 4×1 desired joint position vector $\boldsymbol{q}'_d$ and 4×1 maximum joint velocity vector $\boldsymbol{\omega}'$. The communication buffer ${}^{E,e}b_g$ consists of 8×1 measured joint position vector $\boldsymbol{q}'_m$, and 4×24 measured tactile pressure matrix $\boldsymbol{\psi}$.

The Virtual Effector has two major tasks. It initializes the Real Effector to generate the trajectory in trapezoidal mode and converts the units as follows:

$$\boldsymbol{q}'_d = r_1 \boldsymbol{q}_d, \tag{3}$$

$$\boldsymbol{\omega}' = r_2 \boldsymbol{\omega}, \tag{4}$$

$$\boldsymbol{q}_m = r_3 \boldsymbol{q}'_m, \tag{5}$$

$$\boldsymbol{f} = r_4 \boldsymbol{\psi} \circ \boldsymbol{p}, \tag{6}$$

where r_1, r_2, r_3 and r_4 are constant factors obtained from the BarrettHand gripper documentation and $\boldsymbol{p}$ is a 4×24 matrix of tactile sensors' areas. The $\circ$ operator is the Hadamard product: $(\boldsymbol{A} \circ \boldsymbol{B})_{ij} = A_{ij} B_{ij}$.

The Real Effector is responsible for closed loop position control (with trapezoidal velocity profile) using commanded joint position $\boldsymbol{q}'_d$ and maximum joint velocity $\boldsymbol{\omega}'$.

2.3 Image Recognition and Markers Localization

The system uses special markers attached to the door surface for rough door localization. The Virtual Receptor informs the Control Subsystem about the markers numbers and their poses in camera coordinate system $\boldsymbol{R}$ using the communication buffer ${}^{r,c}b_v$. The Virtual Receptor utilizes the ALVAR library to detect and track artificial markers designed for this library. The Real Receptor transmits images to the Virtual Receptor for further processing using the communication buffer ${}^{R,r}b_v$.

3 Task Algorithm

The task algorithm, which is implemented in the Control Subsystem of embodied agent, is subdivided into two parts: door localization and approaching is the first and actual door opening is the second one. During both motion phases a pose and a stiffness are controlled. Although the Virtual Effector e_m controls pose of $\boldsymbol{W}$, it is possible to control pose of $\boldsymbol{E}$ or $\boldsymbol{C}$ using transformations ${}^{E}_{W}\boldsymbol{T}$ or ${}^{C}_{W}\boldsymbol{T}$.

Two configurations of gripper joints are employed. The first ($\boldsymbol{q}_{door}$) is used for door approach and it is a hook with finger joint angles at 40°, so the distal phalanges are hardly orthogonal to door surface and their artificial skin can detect contact with the door. The second configuration ($\boldsymbol{q}_{handle}$) is proper for the door handle approach and for the door opening and it is a hook with finger joint angles at 75°.

The task algorithm uses the following functions:

- $Trans(\boldsymbol{P})$, where $\boldsymbol{P}$ is a 3×1 vector, returns homogeneous transformation matrix with no rotation for translation from point $[0,0,0]^T$ to $\boldsymbol{P}$,
- $RotZ(\alpha)$ returns homogeneous transformation matrix for rotation by angle α in z axis,
- $RotY(\alpha)$ returns homogeneous transformation matrix for rotation by angle α in y axis,
- $wait(t)$ suspends the algorithm execution by time t,
- $measureEndPosition()$ returns current ${}^{B}_{W}\boldsymbol{T}$ transformation using ${}^{e,c}b_m[\boldsymbol{q}]$ and direct kinematics,
- $measureFingerPosition()$ returns current ${}^{E}_{F}\boldsymbol{T}$ transformation using ${}^{e,c}b_g[\boldsymbol{q}_m]$ and direct kinematics,
- $AddToBuffer(\boldsymbol{b}, \boldsymbol{P})$ increments size of the buffer $\boldsymbol{b}$ and writes point $\boldsymbol{P}$ at its end,
- $atan2(y, x)$ calculates the four-quadrant inverse tangent of $\frac{y}{x}$,
- $max(\boldsymbol{A})$ returns the maximum value in matrix $\boldsymbol{A}$,
- $min(\boldsymbol{v})$ returns the minimum value in vector $\boldsymbol{v}$,
- $maxIndex(\boldsymbol{A})$ returns index of the maximum element in matrix $\boldsymbol{A}$,
- $estimateCircle(\boldsymbol{b})$ returns estimated circle center and circle radius using least squares algorithm and points from buffer $\boldsymbol{b}$

3.1 Door Localization and Approaching

Door and handle approach algorithm uses the *moveRelToMarker* function defined in alg. 1. The function controls the gripper frame $\boldsymbol{E}$ relative to marker's frame $\boldsymbol{M}$. In the line 2 the marker frame is translated to the point $\boldsymbol{P}$ and rotated so that the gripper's orientation is set to door and handle approach as shown in fig. 2(b). In the line 4 the desired transformation for wrist ${}^{B}_{W_d}\boldsymbol{T}$ is calculated on the basis of the following operands: marker's pose relative to robot's base ${}^{B}_{M}\boldsymbol{T}$, the desired gripper's pose relative to marker frame ${}^{M}_{E_d}\boldsymbol{T}$ and the constant wrist-gripper transformation ${}^{E_d}_{W_d}\boldsymbol{T} = {}^{E}_{W}\boldsymbol{T}$.

The transformation from robot's base to the camera is calculated using a visual marker on the wrist with the equation:

$$ {}^{B}_{R}\boldsymbol{T} = {}^{B}_{W}\boldsymbol{T}\, {}^{W}_{W_m}\boldsymbol{T}\, {}^{W_m}_{R}\boldsymbol{T} \tag{7} $$

where ${}^{B}_{W}\boldsymbol{T}$ is calculated from direct kinematics of the manipulator, ${}^{W}_{W_m}\boldsymbol{T}$ is known as a constant transformation from robot's wrist to the wrist marker and ${}^{W_m}_{R}\boldsymbol{T} = {}^{R}_{W_m}\boldsymbol{T}^{-1}$ is wrist marker's pose taken from the Virtual Receptor. The base

Algorithm 1. Move relative to marker procedure

procedure MOVERELTOMARKER($\boldsymbol{P}, t$) ▷ move the gripper to P in marker frame
    ${}^{M}_{E_d}\boldsymbol{T} \leftarrow Trans(\boldsymbol{P})RotY(\pi)RotZ(-\frac{\pi}{2})$ ▷ get homogenous transformation
    ▷ matrix for translation
    ${}^{B}_{W_d}\boldsymbol{T} \leftarrow {}^{B}_{M}\boldsymbol{T}\,{}^{M}_{E_d}\boldsymbol{T}\,{}^{E_d}_{W_d}\boldsymbol{T}$ ▷ calculate desired wrist frame
    ${}^{c,e}b_m[\boldsymbol{r}_t, t] \leftarrow [{}^{B}_{W_d}\boldsymbol{T}, t]$ ▷ send to Virtual Effector
end procedure

Algorithm 2. Door approach algorithm

${}^{R}_{M}\boldsymbol{T} \leftarrow {}^{r,c}b_v[\boldsymbol{T}]$ ▷ get the marker pose from Virtual Receptor
${}^{B}_{M}\boldsymbol{T} \leftarrow {}^{B}_{R}\boldsymbol{T}\,{}^{R}_{M}\boldsymbol{T}$ ▷ calculate marker's pose in base frame
${}^{c,e}b_g[\boldsymbol{q}] \leftarrow \boldsymbol{q}_{door}$ ▷ set the gripper configuration for door approach
${}^{c,e}b_m[\boldsymbol{K}_t, t] \leftarrow [\boldsymbol{k}_{door}, 3s]$ ▷ change the stiffness
$wait(3s)$
$moveRelToMarker(\boldsymbol{P}_s, 5s)$ ▷ move the gripper to the starting pose
$wait(5s)$
$d_{door} \leftarrow 0$
repeat ▷ move the gripper towards door
    $moveRelToMarker(\boldsymbol{P}_s + [0, 0, -d_{door}]^T, 0.125s)$ ▷ move the gripper
    $d_{door} \leftarrow d_{door} + \Delta_{door}$ ▷ decrement desired distance
    $wait(0.1s)$
    $\boldsymbol{f} = {}^{e,c}b_g[\boldsymbol{f}]$ ▷ get tactile force
until $max(\boldsymbol{f}) < f_{threshold}$ ▷ check if contact occured

- camera transformation ${}^{B}_{R}\boldsymbol{T}$ is assumed constant during the algorithm execution until door marker pose is acquired.

The initialization and the door approaching stage is shown in alg. 2. The parameter $\boldsymbol{P}_s$ is the starting point for the gripper relative to the marker pose M. At this stage there are used following other parameters: $\boldsymbol{k}_{door}$ – 6×1 vector of stiffness for the Virtual Effector e_m during the door and handle approach, δ_{door} – a small distance the gripper moves towards the door in every iteration, $f_{threshold}$ – a threshold value for tactile force to check if the contact occured.

Algorithm 3. Hand configuration change

$d_{door} \leftarrow d_{door} - 0.08m$ ▷ increase the distance
$moveRelToMarker(\boldsymbol{P}_s + [0, 0, -d_{door}]^T, 4s)$ ▷ move the gripper
$wait(4s)$
${}^{c,e}b_g[\boldsymbol{q}] \leftarrow \boldsymbol{q}_{handle}$ ▷ set gripper configuration for handle approach
$wait(2s)$
$d_{door} \leftarrow d_{door} + 0.06m$ ▷ decrease the distance
$moveRelToMarker(\boldsymbol{P}_s + [0, 0, -d_{door}]^T, 3s)$ ▷ move the gripper
$wait(3s)$

After the first contact with door, the gripper's configuration is changed (alg. 3). At first, the gripper has to be pulled back to make the finger closing available. The distances $0.08m$ and $0.06m$ (lines 1 and 6) are chosen with respect to direct kinematics of the gripper.

Algorithm 4. Handle approach algorithm

$d_{handle} \leftarrow 0$ ▷ zero distance offset
repeat ▷ move the gripper towards handle
  $moveRelToMarker(\boldsymbol{P}_s + [-d_{handle}, 0, -d_{door}]^T, 0.125s)$ ▷ move the gripper
  $d_{handle} \leftarrow d_{handle} + \Delta_{handle}$ ▷ get closer
  $wait(0.1s)$
  $\boldsymbol{f} = {}^{e,c}b_g[\boldsymbol{f}]$ ▷ get tactile force
until $max(\boldsymbol{f}) < f_{threshold}$ ▷ check if contact occured
${}^{c,e}b_m[\boldsymbol{K}_t, t] \leftarrow [\boldsymbol{k}_{handle}, 3s]$ ▷ change the stiffness
$wait(3s)$
$d_{handle} \leftarrow d_{handle} + r_a$ ▷ push the handle
$moveRelToMarker(\boldsymbol{P}_s + [-d_{handle}, 0, -d_{door}]^T, 5s)$ ▷ move the gripper
$wait(5s)$

After the gripper's configuration is changed, the handle approach is performed (alg.4). At this stage there are following new parameters: δ_{handle} – a small distance the gripper moves towards the handle in every iteration, $\boldsymbol{k}_{handle}$ – 6×1 stiffness vector for the Virtual Effector e_m for pushing the handle. If the contact with handle occurs, the system starts to push the handle, otherwise it finally stops due to kinematics's constraints of the manipulator.

Algorithm 5. Door opening algorithm internal procedures

procedure GETCONTACTPOINTFRAME ▷ get contact point position in frame $\boldsymbol{F}$
  $\boldsymbol{f} \leftarrow {}^{e,c}b_g[\boldsymbol{f}]$ ▷ get tactile force
  ${}^F_C\boldsymbol{T} \leftarrow Trans(\boldsymbol{s}_{maxIndex(\boldsymbol{f})})$ ▷ get contact point homogeneous transformation
  return ${}^F_C\boldsymbol{T}$ ▷ return contact point relative to finger frame
end procedure
procedure GETTRANSFORMATIONS ▷ get actual transformations
  ${}^B_W\boldsymbol{T} \leftarrow measureEndPosition()$ ▷ get wrist pose from direct kinematics
  ${}^E_F\boldsymbol{T} \leftarrow measureFingerPosition()$ ▷ get finger pose from direct kinematics
  ${}^F_C\boldsymbol{T} \leftarrow getContactPointFrame()$ ▷ get contact point pose
  return $[{}^B_W\boldsymbol{T}, {}^E_F\boldsymbol{T}, {}^F_C\boldsymbol{T}]$ ▷ return actual transformations
end procedure

3.2 Door Opening

Door opening algorithm uses the *getContactPointFrame* and *GetTransformations* functions defined in alg. 5. The first one returns the homogeneous transformation matrix ${}^F_C\boldsymbol{T}$ from the contact point frame $\boldsymbol{C}$ to the finger frame $\boldsymbol{F}$.

The orientation of both frames is assumed to be identical. The 24×3 matrix $\boldsymbol{s}$ is tactile geometry and every $i-th$ element $\boldsymbol{s}_i$ is a center point of the $i-th$ tactile sensor relative to the $\boldsymbol{F}$ frame. The *GetTransformations* function returns the current transformations ${}^{B}_{W}\boldsymbol{T}$, ${}^{E}_{F}\boldsymbol{T}$ and ${}^{F}_{C}\boldsymbol{T}$.

Algorithm 6. Door opening algorithm initial motion

1: $[{}^{B}_{W}\boldsymbol{T}, {}^{E}_{F}\boldsymbol{T}, {}^{F}_{C}\boldsymbol{T}] \leftarrow GetTransformations()$ ▷ get current transformations
2: ${}^{E}_{E_d}\boldsymbol{T} \leftarrow Trans(0, r_a, -d_{init})$ ▷ determine gripper's destination pose change
3: ${}^{B}_{C}\boldsymbol{T}^{init}_{d} \leftarrow {}^{B}_{W}\boldsymbol{T}\,{}^{W}_{E}\boldsymbol{T}\,{}^{E}_{E_d}\boldsymbol{T}\,{}^{E_d}_{F_d}\boldsymbol{T}\,{}^{F_d}_{C_d}\boldsymbol{T}$ ▷ calculate contact point's destination pose
4: $\boldsymbol{P}^{init}_{d} \leftarrow {}^{B}_{C}\boldsymbol{T}^{init}_{d}[0,0,0,1]^T$ ▷ save contact point's destination position
5: ${}^{B}_{C}\boldsymbol{T} \leftarrow {}^{B}_{W}\boldsymbol{T}\,{}^{W}_{E}\boldsymbol{T}\,{}^{E}_{F}\boldsymbol{T}\,{}^{F}_{C}\boldsymbol{T}$ ▷ calculate contact point's current pose
6: $\boldsymbol{P}_{contact} \leftarrow {}^{B}_{C}\boldsymbol{T}[0,0,0,1]^T$ ▷ calculate current contact point
7: $AddToBuffer(\boldsymbol{b}, \boldsymbol{P}_{contact})$ ▷ add the current contact point to buffer
8: ${}^{B}_{W_d}\boldsymbol{T} \leftarrow {}^{B}_{W}\boldsymbol{T}\,{}^{W}_{E}\boldsymbol{T}\,{}^{E}_{E_d}\boldsymbol{T}\,{}^{E_d}_{W_d}\boldsymbol{T}$ ▷ calculate desired wrist frame
9: ${}^{c,e}b_m[\boldsymbol{r}_t, t] \leftarrow [{}^{B}_{W_d}\boldsymbol{T}, 3s]$ ▷ send pose and motion time to the Virtual Effector

The initial motion for the door opening is shown in alg. 6. The motion starts from the closed door and ends with the door handle shifted by r_a – value added to the estimated circle radius to push the handle outwards. At this stage the other parameters are: d_{init} – distance of the initial motion in the direction perpendicular to the door's surface. In the line 2, the initial motion's destination pose is calculated. It is the homogeneous transformation matrix ${}^{E}_{E_d}\boldsymbol{T}$ between the current gripper's pose and the desired gripper's pose. In the line 3 the desired contact point frame is calculated and the destination for the contact point is calculated in line 4. Transformations ${}^{E_d}_{F_d}\boldsymbol{T}$ and ${}^{F_d}_{C_d}\boldsymbol{T}$ are constant and and respectively equal to ${}^{E}_{F}\boldsymbol{T}$ and ${}^{F}_{C}\boldsymbol{T}$. The actual contact point is calculated in lines 5 and 6 and it is saved to the empty buffer $\boldsymbol{b}$ in line 7. The desired wrist pose is calculated in line 8.

Algorithm 7. Door opening algorithm - the first kinematic estimation

while not $timeElapsed(3s)$ **do** ▷ repeat this loop for 3 seconds
    $[{}^{B}_{W}\boldsymbol{T}, {}^{E}_{F}\boldsymbol{T}, {}^{F}_{C}\boldsymbol{T}] \leftarrow GetTransformations()$ ▷ get actual position
    ${}^{B}_{C}\boldsymbol{T} \leftarrow {}^{B}_{W}\boldsymbol{T}\,{}^{W}_{E}\boldsymbol{T}\,{}^{E}_{F}\boldsymbol{T}\,{}^{F}_{C}\boldsymbol{T}$ ▷ calculate contact point frame relative to $\boldsymbol{B}$
    $\boldsymbol{P}_{contact} \leftarrow {}^{B}_{C}\boldsymbol{T}[0,0,0,1]^T$ ▷ calculate current contact point
    $AddToBuffer(\boldsymbol{b}, \boldsymbol{P}_{contact})$ ▷ add the current contact point to buffer
    $wait(0.1s)$
end while
$[\boldsymbol{P}_c, r] \leftarrow estimateCircle(\boldsymbol{b})$ ▷ estimate circle using least squares algorithm
$\alpha_{init} \leftarrow atan2(\boldsymbol{b}_{first[y]} - \boldsymbol{P}_{c[y]}, \boldsymbol{b}_{first[x]} - \boldsymbol{P}_{c[x]})$ ▷ calculate door initial angle
$\alpha_{dest} \leftarrow \alpha_{init} + \alpha_{open}$ ▷ calculate door destination angle
$\alpha \leftarrow atan2(\boldsymbol{P}^{init}_{d[y]} - \boldsymbol{P}_{c[y]}, \boldsymbol{P}^{init}_{d[x]} - \boldsymbol{P}_{c[x]})$ ▷ calculate current angle

Acquired contact points are stored in buffer $\boldsymbol{b}$ (alg. 7). After $3s$ the initial motion ends and the points in buffer $\boldsymbol{b}$ are used to estimate a circle, which center $\boldsymbol{P}_c$ corresponds to the door axis and its radius r corresponds to the door handle – axis distance. The first element in the buffer $\boldsymbol{b}$ is $\boldsymbol{b}_{first}$ and the last one is $\boldsymbol{b}_{last}$. The α_{init} is the angle between the door surface and xz plane of base frame $\boldsymbol{B}$ before the initial motion. Analogically, the α_{dest} angle and the α angle relate the vectors affixed to destination and current contact points to xz plane of base frame $\boldsymbol{B}$.

The next door opening stage (alg. 8) begins with the stiffness change (line 1). In each iteration the current door angle based on the current contact point position is calculated (line 5). The desired gripper rotation β around z axis relative to the initial rotation is increased by δ_e (line 7) and limited to actual relative door angle $\alpha_{door} - \alpha_{init}$ (line 8). The desired contact point position $\boldsymbol{P}_d$ (line 9) is calculated from actual estimated circle (line 18) and destination contact point angle α which is increased in every iteration by δ (line 6), where $\delta < \delta_e$. The desired contact point pose ${}^{B}_{C_d}\boldsymbol{T}$ is calculated from the contact point desired frame for the initial motion ${}^{B}_{C}\boldsymbol{T}^{init}_{d}$ translated to the new desired contact point $\boldsymbol{P}_d$ and rotated by the desired gripper rotation β (line 10). The actual contact point $P_{contact}$ is calculated and added to buffer in lines 13, 14 and 15. The wrist destination pose is calculated in line 16, where the transformations ${}^{C_d}_{F_d}\boldsymbol{T}$ and ${}^{F_d}_{W_d}\boldsymbol{T}$ are constant and equal to ${}^{C}_{F}\boldsymbol{T}$ and ${}^{F}_{W}\boldsymbol{T}$.

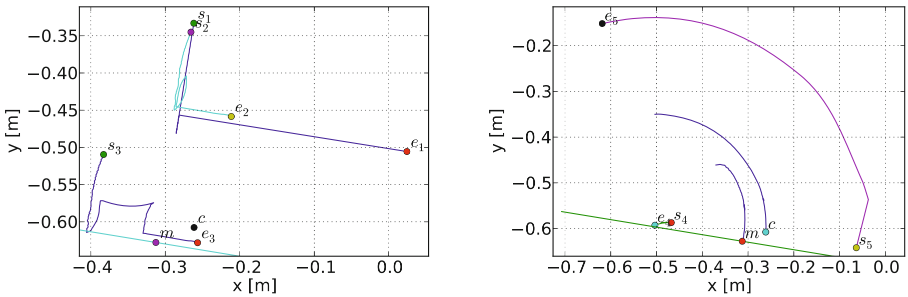
Algorithm 8. Door opening algorithm - door opening

1: ${}^{c,e}b_m[\boldsymbol{K}_t, t] \leftarrow [\boldsymbol{k}_{open}, 3s]$ ▷ change the stiffness
2: $wait(3s)$
3: $\beta \leftarrow 0$
4: **repeat** ▷ open the door
5: $\alpha_{door} \leftarrow atan2(\boldsymbol{b}_{last[y]} - \boldsymbol{P}_{c[y]}, \boldsymbol{b}_{last[x]} - \boldsymbol{P}_{c[x]})$ ▷ calculate current door angle
6: $\alpha \leftarrow \alpha + \delta$ ▷ increase destination angle
7: $\beta \leftarrow \beta + \delta_e$ ▷ increase destination hand rotation angle
8: $\beta \leftarrow min([\beta, \alpha_{door} - \alpha_{init}])$ ▷ limit value to $(\alpha_{door} - \alpha_{init})$
9: $\boldsymbol{P}_d \leftarrow \boldsymbol{P}_c + (r + r_a)[cos\alpha, sin\alpha, 0]^T$ ▷ calculate destination contact point
10: ${}^{B}_{C_d}\boldsymbol{T} \leftarrow Trans(\boldsymbol{P}_d - \boldsymbol{P}^{init}_d)\, {}^{B}_{C}\boldsymbol{T}^{init}_{d} RotZ(-\beta)$ ▷ calculate destination
11: ▷ contact frame
12: $[{}^{B}_{W}\boldsymbol{T}, {}^{E}_{F}\boldsymbol{T}, {}^{F}_{C}\boldsymbol{T}] \leftarrow GetTransformations()$ ▷ get actual position
13: ${}^{B}_{C}\boldsymbol{T} \leftarrow {}^{B}_{W}\boldsymbol{T}\, {}^{W}_{E}\boldsymbol{T}\, {}^{E}_{F}\boldsymbol{T}\, {}^{F}_{C}\boldsymbol{T}$ ▷ calculate current contact frame
14: $\boldsymbol{P}_{contact} \leftarrow {}^{B}_{C}\boldsymbol{T}[0, 0, 0, 1]^T)$ ▷ calculate current contact point
15: $AddToBuffer(\boldsymbol{b}, \boldsymbol{P}_{contact})$ ▷ add the current contact point to buffer
16: ${}^{B}_{W_d}\boldsymbol{T} \leftarrow {}^{B}_{C_d}\boldsymbol{T}\, {}^{C_d}_{F_d}\boldsymbol{T}\, {}^{F_d}_{W_d}\boldsymbol{T}$ ▷ calculate desired wrist frame
17: ${}^{c,e}b_m[\boldsymbol{r}_t, t] \leftarrow [{}^{B}_{W_d}\boldsymbol{T}, 1.0s]$ ▷ send pose to the Virtual Effector
18: $[\boldsymbol{P}_c, r] \leftarrow estimateCircle(\boldsymbol{b})$ ▷ estimate circle using least squares algorithm
19: $wait(0.1)$
20: **until** $\alpha < \alpha_{dest}$ ▷ repeat until the door is open

4 Experiments

Experiments were performed with the presented system and a cabinet shown in fig. 2(b). The real receptor R_v was RGB camera mounted on robot's active head [22]. The experiments were performed for two different distances between door handle and hinges: $0.24m$ and $0.135m$. In the following sections the experimental results are presented for parameters, which led to the successful and robust task execution:

$\boldsymbol{P}_s = [0, -0.1m, 0.3m]^T$, $r_a = 0.25m$, $d_{init} = 0.1m$,
$\alpha_{open} = 110°$, $\delta = 0.005rad$, $\delta_e = 0.04rad$,
$\boldsymbol{k}_{door} = [600\frac{N}{m}, 1000\frac{N}{m}, 1000\frac{N}{m}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}]^T$,
$\boldsymbol{k}_{handle} = [500\frac{N}{m}, 35\frac{N}{m}, 1000\frac{N}{m}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}]^T$,
$\boldsymbol{k}_{open} = [150\frac{N}{m}, 35\frac{N}{m}, 1000\frac{N}{m}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}, 300\frac{Nm}{rad}]^T$,



(a) (b)

Fig. 3. Trajectories registered during the experiment: (a) for the door handle approach, (b) for the door opening, where s is the starting point and e is the end point of a trajectory. Trajectories are labeled as follows: commanded trajectory of the gripper (1), measured trajectory of the gripper (2), measured trajectory of the finger tip (3), trajectory of estimated circle center (4), commanded trajectory of contact point of the finger and the door handle (5). Point m is marker position and the straight line passing point m is a door surface. The line starting from point m is the trajectory of door marker acquired from the Virtual Receptor r_v. Point c is the first contact point of finger and door handle, and the line starting from that point is the trajectory of contact point during door opening.

4.1 Door and Handle Approach

Fig. 3(a) shows trajectories of the gripper and finger tip during door and handle approach. The force measured by tactile sensors during this stage is shown in fig. 4(a) for time $t \in (0, 15)$. The point a is a contact between door surface and finger tip. The gripper is pulled back and the measured force falls. After that, the contact between finger and door handle is measured in point b. The force rises after point b because the gripper trajectory is not canceled immediately.

4.2 Opening the Door

Fig. 3(b) shows trajectories of the gripper and finger tip during door opening. The force measured by tactile sensors during this stage is shown in fig. 4(a) for time $t \in (15, 55)$. Between points c and d the stiffness changes from $\boldsymbol{k}_{door}$ to $\boldsymbol{k}_{handle}$ and the measured force of contact falls down. Between points d and e the gripper is pushed harder towards the handle and the force rises again. After that, between points f and g, the initial door opening motion is performed. Between points h and i the stiffness changes from $\boldsymbol{k}_{handle}$ to $\boldsymbol{k}_{open}$. From point i the door opening motion using circle estimation is performed. Fig. 4(b) shows estimated circle radius during door opening. The fluctuations in estimation are small enough to maintain stable motion.

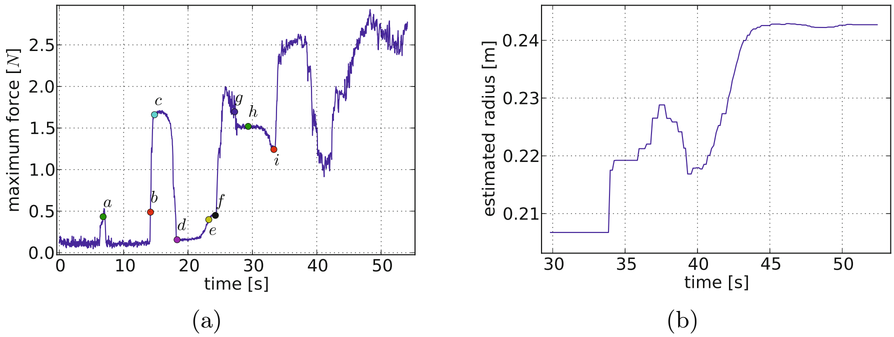


Fig. 4. The experiments: (a) maximum measured finger tip force, (b) estimated circle radius

5 Conclusions

The door approaching and opening algorithm, which has been proposed in this article, is based on the previous work of the authors and current achievements in tactile sensing and impedance control. The algorithm deals with some environment constraints, such as obstacles located in the neighborhood of the door handles or knobs, by maintaining the contact with the single side of the handle. This concept is an alternative for motion generation with force grip or closure grip. Although the controller is stable, a priori parameter determination is needed to maintain contact with handle. Otherwise, for e.g. to high stiffness, the system can fail too execute the task. The future work assumes automated impedance controller parameter learning by analysis of the task execution quality criterion (e.g. composed of contact force oscillation metrics and other statistics). The whole procedure will be verified and generalized for opening the various type of doors (the large doors will be opened with active torso and mobile base) and drawers. Finally, the manipulation system behavior will be analyzed in cooperation with the visual subsystem, which will perform door and handle detection and localization instead of markers detection.

Acknowledgment. This project was funded by the National Science Centre according to the decision number DEC-2012/05/D/ST6/03097. Tomasz Winiarski has been supported by the European Union in the framework of the European Social Fund through the Warsaw University of Technology Development Programme.

References

1. Stefańczyk, M., Walęcki, M.: Localization of essential door features for mobile manipulation. In: Szewczyk, R., Zieliński, C., Kaliczyńska, M. (eds.) Recent Advances in Automation, Robotics and Measuring Techniques. AISC, vol. 267, pp. 487–496. Springer, Heidelberg (2014)
2. Meeussen, W., Wise, M., Glaser, S., Chitta, S., McGann, C., Mihelich, P., Marder-Eppstein, E., Muja, M., Eruhimov, V., Foote, T., et al.: Autonomous door opening and plugging in with a personal robot. In: 2010 IEEE International Conference on Robotics and Automation (ICRA), pp. 729–736. IEEE (2010)
3. Zieliński, C., Winiarski, T.: Motion generation in the MRROC++ robot programming framework. International Journal of Robotics Research 29(4), 386–413 (2010)
4. Zieliński, C., Winiarski, T., Mianowski, K., Rydzewski, A., Szynkiewicz, W.: End-effector sensors' role in service robots. In: Kozłowski, K. (ed.) Robot Motion and Control 2007. LNCIS, vol. 360, pp. 401–414. Springer, Heidelberg (2007)
5. Staniak, M., Winiarski, T., Zieliński, C.: Parallel visual-force control. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2008 (2008)
6. Winiarski, T., Banachowicz, K., Stefańczyk, M.: Safe strategy of door opening with impedance controlled manipulator. Journal of Automation Mobile Robotics and Intelligent Systems 7(4), 21–26 (2013)
7. Karayiannidis, Y., Smith, C., Vina, F.E., Ogren, P., Kragic, D.: "open sesame!" adaptive force/velocity control for opening unknown doors. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4040–4047. IEEE (2012)
8. Winiarski, T., Banachowicz, K.: Opening a door with a redundant impedance controlled robot. In: 9th Workshop on Robot Motion & Control (RoMoCo), pp. 221–226 (2013)
9. Ott, C., Bäuml, B., Borst, C., Hirzinger, G.: Employing cartesian impedance control for the opening of a door: A case study in mobile manipulation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems Workshop on Mobile Manipulators: Basic Techniques, New Trends & Applications (2005)
10. Katz, D., Kazemi, M., Andrew Bagnell, J., Stentz, A.: Interactive segmentation, tracking, and kinematic modeling of unknown 3d articulated objects. In: 2013 IEEE International Conference on Robotics and Automation (ICRA), pp. 5003–5010. IEEE (2013)
11. Ruhr, T., Sturm, J., Pangercic, D., Beetz, M., Cremers, D.: A generalized framework for opening doors and drawers in kitchen environments. In: International Conference on Robotics and Automation (ICRA), pp. 3852–3858. IEEE (2012)
12. Chung, W., Rhee, C., Shim, Y., Lee, H., Park, S.: Door-opening control of a service robot using the multifingered robot hand. IEEE Transactions on Industrial Electronics 56(10), 3975–3984 (2009)

13. Winiarski, T., Woźniak, A.: Indirect force control development procedure. Robotica 31, 465–478 (2013)
14. Schmid, A., Gorges, N., Goger, D., Worn, H.: Opening a door with a humanoid robot using multi-sensory tactile feedback. In: International Conference on Robotics and Automation (ICRA), pp. 285–291. IEEE (2008)
15. Kornuta, T., Zieliński, C.: Robot control system design exemplified by multi-camera visual servoing. Journal of Intelligent & Robotic Systems, 1–25 (2013)
16. Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y.: Ros: an open-source robot operating system. In: ICRA Workshop on Open Source Software, vol. 3 (2009)
17. Zubrycki, I., Granosik, G.: Test setup for multi-finger gripper control based on robot operating system (ros). In: 2013 9th Workshop on Robot Motion and Control (RoMoCo), pp. 135–140. IEEE (2013)
18. Bruyninckx, H.: Open robot control software: the orocos project. In: International Conference on Robotics and Automation (ICRA), vol. 3, pp. 2523–2528. IEEE (2001)
19. Albu-Schäffer, A., Ott, C., Hirzinger, G.: A unified passivity-based control framework for position, torque and impedance control of flexible joint robots. The International Journal of Robotics Research 26(1), 23–39 (2007)
20. Caccavale, F., Natale, C., Siciliano, B., Villani, L.: Six-dof impedance control based on angle/axis representations. IEEE Transactions on Robotics and Automation 15(2), 289–300 (1999)
21. Albu-Schaffer, A., Ott, C., Frese, U., Hirzinger, G.: Cartesian impedance control of redundant robots: Recent results with the dlr-light-weight-arms. In: International Conference on Robotics and Automation (ICRA), vol. 3, pp. 3704–3709. IEEE (2003)
22. Walęcki, M., Stefańczyk, M., Kornuta, T.: Control system of the active head of a service robot exemplified on visual servoing. In: 9th Workshop on Robot Motion and Control (RoMoCo), pp. 48–53 (2013)