INTERACTION OF A FLEXIBLE ROBOT WITH ITS ENVIRONMENT

Michael Kastner, Hubert Gattringer, Hartmut Bremer
Institute for Robotics, Johannes Kepler University, Altenberger Strasse 69, 4040 Linz, Austria

Martin Ramsauer, Paolo Ferrara
FerRobotics Compliant Robot Technology GmbH, Altenberger Strasse 69, 4040 Linz, Austria

Keywords: Fludic Muscles, Pneumatic Robot, Contact Detection, Interactive Programming, Human Robot Interaction.

Abstract: Uncomplicated and safe programming interfaces as well as flexible programs themselves become important when robots are used for small lot size tasks or are operated by personnel without special robotics education. This work takes a look at safe and easy interaction of a flexible articulated robot arm – actuated by fluidic muscles – with its environment. A contact detection scheme for stiff collisions at speeds between 50 and 250 mm/s is presented and measurement results are disucussed. Moreover, a programming by demonstration concept is described on the basis of a pick and place task. Both strategies (implemented on a seven axis handling robot) rely on physical models to allow an operation without extra sensors.

1 INTRODUCTION

Programming a robot requires a high level of subject-specific knowledge and concentration. Human errors can cause great damage to the robots environment – including people. Studies showed that most robot related accidents occur during programming or fault recovery (Clark and Lehto, 1999). For small lot sizes, frequent reprogramming and adaptation increases the occurance of these scenarios, thus heightening the overall probability of harm to process equipment and personnel.

Fluidic muscles are interesting actuators for devices interacting with humans, because of their compliance. See (Daerden and Lefeber, 2002) for an technology overview and (Van Damme et al., 2005) for more on soft manipulators. Besides safety considerations covered in (Bicchi and Tonietti, 2004), this compliance can also be used to let the robot be more aware of its surroundings.

For this paper we consider a robot actuated by commercially available muscles (Hesse, 2003) of the McKibben type. We show how the physical model of the robot and its actuators can be used in two scenarios of robot environment interaction. This is done without the need for any sensors in addition to the ones present in the control loops (see Figure 1b).

In the first setting, the robot moves along a trajectory at rather low speed and should be able to detect a (stiff) collision with its surrounding at the tool without causing damage. This sense of touch is meant to free the programmer from the necessity to provide the exact data of the process setup to the robot. Measurement results for different speeds and spatial directions are discussed.

The second part describes the direct interaction of the operator with the robot to allow the programming of a simple pick and place task completely by demonstration.

2 SYSTEM ARCHITECTURE

The robot configuration (1a) comprises seven rotational joints. Five are actuated pneumatically. Of these, three (1, 2 and 4) are driven by the one degree of freedom muscle setup detailed below. The hand joint (6, 7) is operated by a combination of three fluidic muscles and a cardan mechanism. The remaining two joints (3, 5) contain standard brushless DC motors with harmonic drive gearing.
As shown in Figure 1b, a pneumatic muscle drive consists of two muscles (Figure 1c) in an antagonistic setup connected by a sprocket chain. The linear mo-

Kastner M., Gattringer H., Bremer H., Ramsauer M. and Ferrara P. (2010).
INTERACTION OF A FLEXIBLE ROBOT WITH ITS ENVIRONMENT.
In *Proceedings of the 7th International Conference on Informatics in Control, Automation and Robotics*, pages 230-233
DOI: 10.5220/0002918402300233

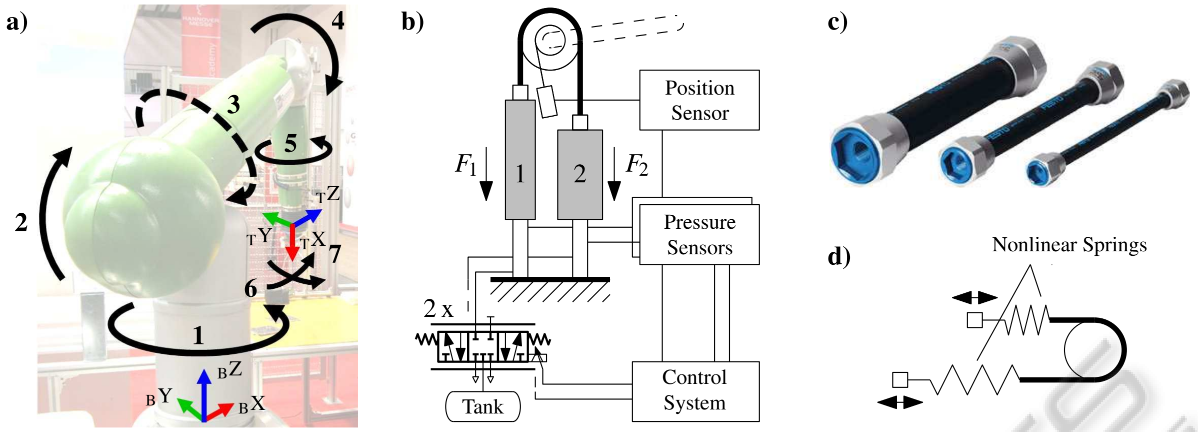

Figure 1: Kinematic setup (joints, base and tool coordinate systems) of the considered robot *R*omo (a), mostly actuated by pneumatic drives like (b). The drives utilize fluidic muscles (c) and their characteristic is similar to the scheme shown in (d).

tion of the actuator is transformed into a rotation by an according sprocket wheel supported by a ball bearing. The absolute rotation angle is measured by a cable extension sensor. Airflow to and from the muscles is regulated by proportional directional valves. The air pressure inside each actuator is measured by a sensor mounted near the inlet. All sensors are connected to 16 bit A/D converters.

The compliance of the system stems from the soft muscles and the compressible work medium air. In this respect the actuation is roughly equivalent to two nonlinear springs with adjustable pre-tensioning (see Figure 1d).

3 MODELING

3.1 Fluidic Muscles

The static relations between force F, contraction h and pressure p of the fluidic muscles (Festo MAS, Figure 1c) are provided in the form of datasheet diagrams by the manufacturer. For this work, we used approximations of the form

$$F = a(h)\,p + b(h)\,, \qquad (1)$$

where $a(h)$ and $b(h)$ are polynomials of order three and six. Their coefficients have been identified in experiments. One model muscle was used for each different diameter.

When two fluidic muscles are combined into an antagonistic setup, the resulting actuator torque is

$$Q_{\mathrm{M}}(\Delta p, q) = r_{\mathrm{S}}\,(F_2 - F_1)\,, \qquad (2)$$

with the sprocket wheel radius r_{S} and the muscle forces F_1 and F_2 (see Figure 1b). Q_{M} can be written as a function of the pressure difference $\Delta p = p_2 - p_1$ and the joint angle q, to which the contractions h_1 and h_2 are geometrically linked.

3.2 Multi Body System

For modeling the mechanical part of the robot, the equation of motion of the multibody system

$$\mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{Q}_{\mathrm{M}} \qquad (3)$$

is used. Here, $\mathbf{q}$ and its time derivatives are the vectors of the joint angles, velocities and accelerations respectively, $\mathbf{M}$ is the mass matrix, $\mathbf{g}$ a term that includes gravitational forces, coriolis forces and so on. A calibration term is also part of $\mathbf{g}$. The entries of the vector $\mathbf{Q}_{\mathrm{M}}$ are the torques of the joint actuators (mostly muscle pairs like in eq. 2). We calculated and implemented this model by using the projection equation in subsystem formulation (Bremer, 2008)

$$\sum_{k=1}^{N}\left(\frac{\partial \dot{\mathbf{y}}_k}{\partial \dot{\mathbf{q}}}\right)^{\mathrm{T}}\left(\mathbf{M}_k(\mathbf{y}_k)\,\ddot{\mathbf{y}}_k + \mathbf{g}_k(\mathbf{y}_k,\dot{\mathbf{y}}_k) - \mathbf{Q}_k\right) = \mathbf{0} \qquad (4)$$

which allowed us to combine smaller segments k (one drive and the attached arm each) of the robot in a modular way. $\mathbf{y}$ are describing coordinates used for the separate subsystems while the rest of the notation is analogous to the one used in Equation 3. Geometry and inertia data were exported from the CAD construction files, damping terms were identified on the real system.

4 CONTACT DETECTION

A robot manipulator is moving along some path (Figure 2). At an unknown location, it will collide with its

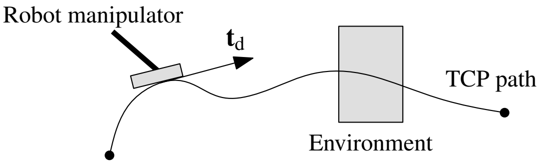

Figure 2: Contact detection scenario.

surrounding. This is deliberate but must be handeled in a manner as not to inflict damage. An algorithm should be found to allow the robot an autonomous (i.e. without the use of any extra sensors) detection of such a collision.

After the detection, the robot should abort the maneuver, report the incident to the sequence control and hold its current position. Due to the limited bandwidth of the pneumatic robot system, it is impossible to avoid the peak in the contact force upon collision. This has to be dealt with by choosing an appropriate approaching velocity. After the impact, the robot should not push against the surrounding.

The focus for this task is on two stiff clashing objects. (Haddadin et al., 2008) cover similar topics for an electrically actuated robot with focus on safety.

4.1 Contact Force Estimation

In the contact case, the model multibody dynamics Equation 3 does not hold any longer. There is a remaining term

$$\mathbf{Q}_\mathrm{C} = \mathbf{M}(\mathbf{q})\,\ddot{\mathbf{q}} + \mathbf{g}(\mathbf{q},\dot{\mathbf{q}}) - \mathbf{Q}_\mathrm{M}(\mathbf{q},\mathbf{p}) \tag{5}$$

that we interpret as a vector of external contact torques acting on the joints. Deviations from the model are also included there as well as measurement errors. The velocities $\dot{\mathbf{q}}$ and accelerations $\ddot{\mathbf{q}}$ are estimated from the position sensor data.

As already mentioned, we are interested in the tool contact scenario. Therefore, we use the relation for the virtual work

$$\delta\mathbf{x}_\mathrm{TCP}^\mathrm{T}\,\mathbf{F}_\mathrm{C} = \delta\mathbf{q}^\mathrm{T}\,\mathbf{Q}_\mathrm{C} \tag{6}$$

and

$$\delta\mathbf{x}_\mathrm{TCP} = \frac{\partial\mathbf{x}_\mathrm{TCP}}{\partial\mathbf{q}}\,\delta\mathbf{q} = \frac{\partial\mathbf{v}_\mathrm{TCP}}{\partial\dot{\mathbf{q}}}\delta\mathbf{q} \tag{7}$$

to get an equivalent tool center force estimation,

$$\mathbf{F}_\mathrm{C} = \left(\left(\frac{\partial\mathbf{v}_\mathrm{TCP}}{\partial\dot{\mathbf{q}}}\right)^\mathrm{T}\right)^{-1}\mathbf{Q}_\mathrm{C}. \tag{8}$$

The vector $\mathbf{x}_\mathrm{TCP} = [x\ y\ z]^\mathrm{T}$ contains the cartesian coordinates of the tool center point, $\mathbf{v}_\mathrm{TCP}$ is the according absolute translational speed. From the kinematics of the robot (Figure 1a), one can see that for a reasonable tool geometry, joints 6 and 7 are insensitive to contact forces because of the small lever arms. More noise than information would be added, which is why we did not include these joints in our algorithm. Joints 3 and 5 are also excluded as they are not actuated pneumatically. The Jacobian using the velocities then reads

$$\frac{\partial\mathbf{v}_\mathrm{TCP}}{\partial\dot{\mathbf{q}}} = \begin{bmatrix} \frac{\partial v_\mathrm{x}}{\partial\dot{q}_1} & \frac{\partial v_\mathrm{x}}{\partial\dot{q}_2} & \frac{\partial v_\mathrm{x}}{\partial\dot{q}_4} \\ \frac{\partial v_\mathrm{y}}{\partial\dot{q}_1} & \frac{\partial v_\mathrm{y}}{\partial\dot{q}_2} & \frac{\partial v_\mathrm{y}}{\partial\dot{q}_4} \\ \frac{\partial v_\mathrm{z}}{\partial\dot{q}_1} & \frac{\partial v_\mathrm{z}}{\partial\dot{q}_2} & \frac{\partial v_\mathrm{z}}{\partial\dot{q}_4} \end{bmatrix}. \tag{9}$$

4.2 Detection Criterion

We base the decision whether a tool impact occured or not on the two basic ideas that in the contact case

1. the model Equation 3 changes to Equation 5 and
2. the controller (which was designed for free trajectories) performance declines significantly resulting in a limited tracking accuracy.

The most simple way to utilize the first idea is to compare the projected contact force

$$F_\mathrm{C} = \mathbf{F}_\mathrm{C}\cdot\mathbf{t}_\mathrm{d} \tag{10}$$

($\mathbf{t}_\mathrm{d}$ is the tangent vector of the desired path, Figure 2) to some threshold L in the form

$$C = \begin{cases} 1 & \text{if } F_\mathrm{C} > L \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

where $C = 1$ means that contact is detected. Practically, it was impossible to find an L value resulting in a low amount of false positives and negatives over the desired workspace and speed range. This seems to stem from the fact that the actuator model described in Section 3.1 neglects too many effects (like drift and hysteresis) to provide an accurate force measure.

To just detect the impact, we modified the criteron to use

$$\dot{F}_\mathrm{C}(\mathrm{s}) = \frac{\mathrm{s}}{(\mathrm{s}+\lambda)^2}\,F_\mathrm{C}(\mathrm{s}), \tag{12}$$

an estimation of the force derivative with some additional low-pass filtering applied.

We also conducted experiments with the second aforementiond idea. Including the controller error on acceleration level in the form

$$D = \dot{F}_\mathrm{C}\left(\frac{(\ddot{\mathbf{x}}_\mathrm{d}-\ddot{\mathbf{x}})}{\|\dot{\mathbf{x}}\|}\cdot\mathbf{t}_\mathrm{d}\right) \tag{13}$$

showed the best results when again compared to some constant level L. Index d denotes desired values. For very low speeds (below 50 mm/s) the only robust detection found was via the position error $(\mathbf{x}_\mathrm{d}-\mathbf{x})$ which led to a rather big delay in the detection. In most cases, this will render the detection useless as such slow movements are typically used for delicate handling scenarios.

4.3 Measurement Results

We evaluated the detection by driving the manipulator against a workbench at various speeds (50, 100, 150 and 250 mm/s), in different directions (X+, Y+, Z- – see Figure 1a – and mixed diagonal) and with 0.5 or 5 kg of payload mass. Each combination was measured ten times.

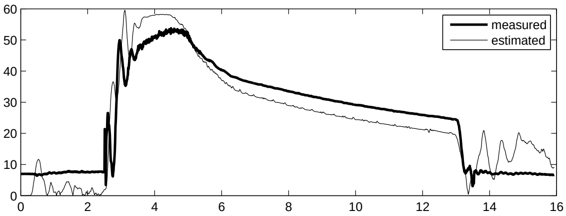

Figure 3: Example of estimated normal contact force in N over time in s – compared to external sensor measurement.

Although the force characteristic (measured with a three axis force sensor) varied over the different scenarios, the detection was successful for nearly all cases without adapting the threshold. Most problematic was the combination of low speed and small payload. Here we found at the same time false negatives (5 for X+) and positives (3 for Y+). Other than that, only one detection of over 300 failed.

The contact force – excluding impact peaks – always stayed below 20 N (and mostly below 10 N for speeds below 150 mm/s).

5 SHOW-DO PROGRAMMING

In guidance mode, a human can grasp the robot and move it freely through the workspace. The robot is still supported by the torques resulting from the model in Equation 3 but any additional effort from the controller is limited to very low values. Due to the soft joints, one can interact with the robot along the complete structure.

The guidance mode can be used to quickly move the robot out of the way, to interactively teach positions or to record complete continuous path segments. In our showcase we used a handle with two buttons, mounted on the lower arm of the robot, to let the user teach complete pick and place applications – similar to a macro recorder known from personal computer software. Tool actions trigger special behaviour – for example the ”pick” macro at execution time moves the manipulator in tool direction until it touches the workpiece and does not solely rely on the recorded position information.

The interaction with the robot proved to be intuitive and all kinds of people were quickly able to perform programming tasks.

6 CONCLUSIONS

In this paper we show how to use models of the mechanical system and the actuators of a pneumatically driven robot for interaction with the environment. The illustrated approach worked well when the human operator compensated the remaining model uncertainties. The completely autonomous interaction also showed good results for the simple case of contact detection.

A first effort to employ the presented interaction possibilities for more intuitive programming led to promising feedback from users.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Austrian Center for Competence in Mechatronics (ACCM) for their support.

REFERENCES

Bicchi, A. and Tonietti, G. (2004). Fast and ”soft-arm” tactics [robot arm design]. *Robotics & Automation Magazine, IEEE*, 11(2):22–33.

Bremer, H. (2008). *Elastic Multibody Dynamics: A Direct Ritz Approach*, chapter 4: Rigid Multibody Systems, pages 59–113. Springer-Verlag GmbH.

Clark, D. R. and Lehto, M. R. (1999). *Handbook of Industrial Robotics*, chapter 36: Reliability, Maintenance and Safety of Robots, pages 717–754. John Wiley & Sons, 2 edition.

Daerden, F. and Lefeber, D. (2002). Pneumatic artificial muscles: Actuators for robotics and automation. *European journal of mechanical and environmental engineering*, 47:11–21.

Haddadin, S., Albu-Schäffer, A., De Luca, A., and Hirzinger, G. (2008). Collision detection and reaction: A contribution to safe physical human-robot interaction. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3356–3363.

Hesse, S. (2003). *The Fluidic Muscle in Application*. Festo AG & Co.KG.

Van Damme, M., Daerden, F., and Lefeber, D. (2005). A pneumatic manipulator used in direct contact with an operator. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*.