Mobile Manipulation with a Kinematically Redundant Manipulator for a Pick-and-Place Scenario

Berntorp, Karl; Årzén, Karl-Erik; Robertsson, Anders

Published in:
[Host publication title missing]

2012

2012 IEEE International Conference on Control Applications (CCA)
Part of 2012 IEEE Multi-Conference on Systems and Control
October 3-5, 2012. Dubrovnik, Croatia

Mobile Manipulation with a Kinematically Redundant Manipulator for a Pick-and-Place Scenario

Karl Berntorp, Karl-Erik Årzén, and Anders Robertsson

Abstract— **Mobile robots and robotic manipulators have traditionally been used separately performing different types of tasks. For example, industrial robots have typically been programmed to follow trajectories using position sensors. If combining the two types of robots and adding sensors new possibilities emerge. This enables new applications, but it also raises the question of how to combine the sensors and the added kinematic complexity.**

An omni-directional mobile robot together with a new type of kinematically redundant manipulator for future use as a service robot for grocery stores is proposed. The scenario is that of distributing groceries on refilling shelves, and a constraint-based task specification methodology to incorporate sensors and geometric uncertainties into the task is employed. Sensor fusion is used to estimate the pose of the mobile base online. Force sensors are utilized to resolve remaining uncertainties. The approach is verified with experiments.

I. INTRODUCTION

Mobile platforms are used in industry for a variety of applications, one example being transporting goods in warehouses. Typically the environment is well defined to make the robot's interaction with its surroundings as predictable as possible.

When it comes to manipulation the robots are usually fixed to the ground, programmed to follow predefined trajectories using position control for static environments. This works well with high accuracy in a structured and predefined environment, but renders problems as soon as uncertainties are introduced in the task.

Since a number of years the attention has been drawn to combine mobile robots with manipulators to open up new potential applications. For many new applications, such as in service robotics, there is a need for the robots to work autonomously in an uncertain environment. There are several challenges when it comes to mobile manipulation. For example, the question of how to coordinate the movements between the mobile base and the manipulator is a nontrivial issue. Another issue is how to define and specify the control task.

The task in this paper is to use a new type of light-weight two-armed industrial robot combined with a mobile omni-directional base to coordinate movements, where the coordination involves distributing groceries on the mobile base while it is moving. The long term objective is a grocery store robot localizing and picking up items on shelves using vision for object recognition. Sensor fusion from wheel encoders, a camera, and an inertial measurement unit is used to estimate the pose of the mobile base. Because of different sampling rates and computational demands some of the measurements will arrive delayed, often referred to as out-of-sequence measurements (OOSM). The delays will have to be accounted for if high precision is wanted. An estimation algorithm based on the OOSMs approach is therefore employed to estimate the robot's pose, see [1], [2]. The manipulator uses force sensors together with impedance control, see [3], which increases the robustness with respect to unmodeled disturbances. To incorporate sensors, introducing uncertainties such as the pose estimates into the control problem, and to construct a framework for coordinated movements the constraint-based task specification methodology (iTaSC) in [4] is exploited. As will be demonstrated in the coming sections it is an approach well suited for handling robot systems with redundant degrees of freedom.

The iTaSC methodology has previously mostly been used for stationary manipulators. Although an example of applying iTaSC on mobile robots is found in [4] it is a very simplistic example.

A. Related Work

An example of motion coordination is given in [5], where the coordination problem is formulated as a nonlinear optimization problem. The result is evaluated on a two degrees-of-freedom mobile base combined with a three degrees-of-freedom manipulator. Other examples of motion coordination are [6], [7], where the coordination is solved by viewing the mobile base as the mechanism with coarse and slow dynamics, and the manipulator is the fast and accurate device. The coordination between the two devices is then done considering internal forces. In [8] the problem of door opening is considered. The mobile base motion is independent of the arm motion, whereas the arm motion is coupled to the mobile base through sensing of the reaction forces of the environment. Another work is [9], where a behavior-based system for controlling the platform is integrated with a hybrid dynamic system for the manipulator control.

An early framework for specifying end-effector tasks is known as the operational space formulation, see [10]. For general tasks that involve end-effector motion and contact forces, generalized task specification matrices are introduced to facilitate modeling. A work on specification of tasks for force control is [11], where a theory of force control based on models of the task geometry is introduced.

K. Berntorp, karl.berntorp@control.lth.se, K-E. Årzén, and A. Robertsson are with the Department of Automatic Control, Lund University, Lund, Sweden

B. Outline

The experimental setup and pick-and-place scenario are introduced in Section II. The task modeling is explained in Section III. A step-by-step description of the pick-and-place scenario is found in Section IV. The experimental results are shown in Section V. Finally, the conclusion and future works are in Section VI.

II. EXPERIMENTAL SETUP

A. Robot System

The robot system used was composed of a four-wheeled omni-directional mobile robot equipped with eight motors, two for each wheel, together with the new concept robot from ABB named FRIDA, see [12]. The mobile robot, which was built at IPA Fraunhofer in Stuttgart, was previously used in the DESIRE project, see [13]. For the experiments in this work it was equipped with a six degrees-of-freedom IMU from Xsens, see [14] for more information, aligned with the coordinate frame of the robot. A calibration procedure was used to calibrate for imperfections in the physical alignment of each component, gains, offsets and temperature relations. The IMU provided measurements with a rate of 100 (Hz). Using the calibration the accelerometer and gyro vectors, expressed in the IMU's local coordinate frame, were computed using an onboard processor. The wheel encoder measurements were the basis of the velocity measurements; based on the kinematics of the robot the velocity vector was extracted with a rate of 20 (Hz). The mobile base used ROS, see [15], for control and navigation.

FRIDA, see Fig. 1, is a dual-arm lightweight manipulator. Both arms have seven degrees of freedom, which means that they have one redundant degree of freedom each. The robot is designed to be intrinsically safe, which is accomplished by having low payload and robot inertia, a mechanical design free from sharp edges, as well as covering exposed regions with soft padding. Also, power and speed limitations together with collision detection are implemented. The robot is controlled with the ABB IRC5 robot control system, which has been extended with an external control system, see [16] and [17], that makes it possible to alter the references for the low-level joint velocity and position loops. The external controllers were built in Matlab/Simulink. The Real-Time workshop toolbox was used for code generation, and the program was run on a Linux Xenomai PC with communication between the IRC5 control system and the external control system via a dedicated Ethernet connection. For communication between the two robots Java with PalCom, see [18], was used.

A two-fingered vacuum tooling was used to grip items. Since the gripper only has two fingers care has to be taken when designing the strategy so that the gripper does not lose the item because of external forces.

For time being FRIDA is not mounted on the mobile base, nor does it have a wrist-mounted force/torque sensor. Hence, to perform pick-and-place operations a six degrees-of-freedom ATI Mini40 force/torque sensor was mounted on the mobile base. On top of the force sensor a plate with elevated borders was placed to simulate a shelf with walls, see Fig. 2 for a picture of the setup. That the mobile base and the manipulator are not rigidly connected, and that the force sensor is not yet mounted on the manipulator's wrist is not any real limitation. Because of iTaSC the kinematic calculations are easily adapted to that scenario. Also, because of the robustness of the mobile base the dynamics should not influence the controllers in an undesired way.

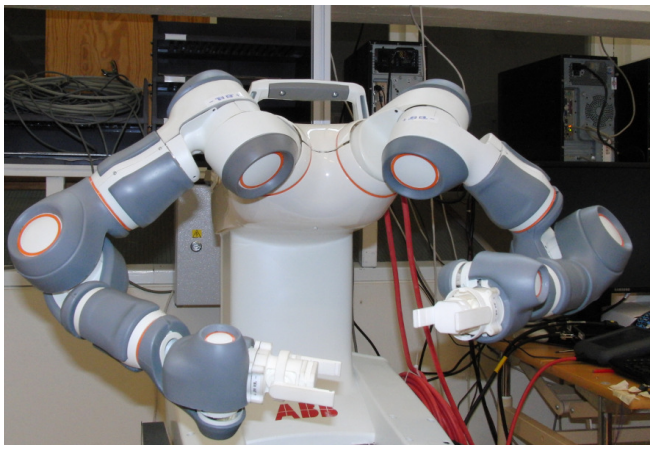

Fig. 1. The new ABB concept robot FRIDA used in the experiments. A two-fingered vacuum tooling is attached to each wrist.

A roof mounted camera was situated above the robots' workspace. Using the camera the absolute position and orientation of the mobile robot was calculated. The algorithms for object tracking and feature detection fall outside the scope of this article, but are surveyed in [19]. For an introduction to inertial and visual sensing see [20]. In this work the vision algorithm gave position measurements with a frame rate of approximately 3 (Hz).

Note that the vision algorithm handles frame rates up to 25 (Hz), but sometimes, because of limited communication or computation resources, it is advantageous to be able to run at lower frame rates and still have high precision. Therefore the low frame rate should be seen as a robustness measure of the control loops. Furthermore, the vision algorithm provides a timestamp. To get correct and robust estimates a pose estimation algorithm has to consider the time delay that the vision algorithm gives rise to. This will be discussed briefly in Section III-D. The vision algorithm as well as the pose estimation algorithm were implemented in Matlab.

B. Pick-and-Place Scenario

The scenario considered is the following: The manipulator should first pick up a can positioned at a fixed known position. The size of the can may be unknown, but it should of course fit into the gripper's fingers. The shape of the can is cylindrical. The manipulator should be able to place the item at the corner of the plate while the mobile base drives around. Furthemore, it should be able to place several cans next to each other. The size of the can is considered unknown up to a couple of centimeters in each direction, so force control is used to ensure that the item is placed in the correct position.

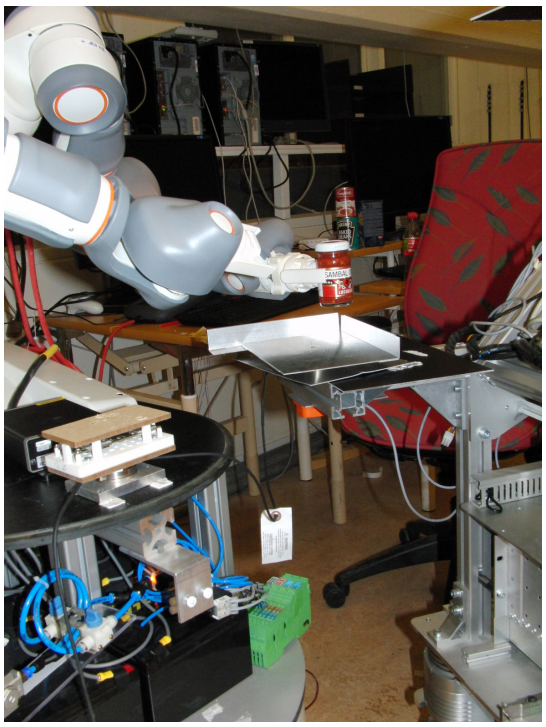

Fig. 2. The setup for the experiments. Under the shelf, seen in the middle of the figure, a force sensor is mounted. The orange hull visible under the force sensor is the IMU. Note that only the mobile base and the arm holding the grocery are used in this setup.

III. MODELING

A. Constraint-Based Task Specification

As mentioned previously, for task modeling the constraint-based task specification framework is used. Since a thorough discussion about this framework already exists in [4], only a summary of the parts and extensions used for this work will be given here.

The constraint-based task specification framework specifies the relative motion of objects by introducing constraints. Constraints can express geometric relationships, force relationships, velocity relationships, or some other relationships. These constraints are specified using kinematic chains. Normally a kinematic chain contains two object frames, $o1$ and $o2$, and two feature frames, $f1$ and $f2$. The object frames are rigidly attached to the manipulated object and the object that manipulates. The feature frames should be attached in such a way that they simplify the problem of specifying constraints that define the task. Furthermore, they should be linked to $o1$ and $o2$, respectively. The different transformations between the frames may either be constant or nonconstant. In total there are six degrees of freedom distributed over the transformations, and these are represented by the feature coordinates χ_f. The feature coordinates are usually partitioned as $\chi_f = (\chi_{fI} \quad \chi_{fII} \quad \chi_{fIII})^{\mathrm{T}}$, where χ_{fI} represents the relative motion of f_1 with respect to $o1$, χ_{fII} represents the relative motion of $f2$ with respect to $f1$, and χ_{fIII} represents the relative motion of $o2$ with respect to $f2$. To obtain the feature coordinates parts of the inverse kinematics of the kinematic chains have to be solved.

Normally not all transformations are exactly known, implying that some uncertainties exist. These uncertainties are modeled by introducing auxiliary transformations, placed between the frames where the uncertainties occur. The degrees of freedom of the uncertainties are given by χ_u, the uncertainty coordinates. For example, frame $o1'$ could model the uncertainty of $o1$, where the degrees of freedom between the two frames are represented by χ_u.

The variables that are of interest to constrain are chosen by specifying outputs y. These can, in general, be functions of the feature and joint coordinates, but often the kinematic chain is chosen such that y directly corresponds to some or all of the feature coordinates χ_f. If fewer outputs are chosen than the degrees of freedom in the robot system the system will be underconstrained, and then the redundancy can be utilized to perform an additional task— for example, minimizing the norm of the joint velocities. It could also be used for emphasizing that not all joints are equally important, since sometimes it may be advantageous to concentrate most of the motion to either the mobile base or the manipulator.

B. Kinematic Chain and Task Specification for Pick-and-Place Scenario

One kinematic chain is used to model the task, and the resulting object and feature frames are shown in Fig. 3. For further clarification of how the feature and uncertainty coordinates connect the different transformations, see Fig. 4.:

- The world frame, w, is rigidly connected to the base of the manipulator.
- Frame $o1$ is fixed to the mobile base with its z-axis along the world z-axis. It is related to w by the traversed path of the mobile base (i.e., q_{mob}). The unknown position of the mobile base is modeled by $o1'$.
- Frame $o2$ is fixed to the flange of the manipulator. It is related to w by the kinematics of the robot (i.e., q_{man}).
- Frame $f1$ is located at one of the corners of the mobile base's plate. It is related to $o1$ by a constant translation.
- Frame $f2$ is connected to the tool center point, related to $o2$ by a constant translation. The unknown length of the gripped item is modeled by $f2'$.

The feature coordinates are collected into the transformation between $f1$ and $f2$ as

$$\chi_{fII} = (x \quad y \quad z \quad \psi \quad \theta \quad \phi)^{\mathrm{T}},$$

where y should be interpreted as translation in the y-direction. Since all degrees of freedom are collected between $f1$ and $f2$, both χ_{fI} and χ_{fIII} are zero-dimensional.

The first three feature coordinates are cartesian translations along the axes of $f1$, and the last three are rotational coordinates parametrized by ZYX (yaw-pitch-roll) Euler angles. The uncertainties in the task include, as already mentioned, the three-dimensional pose of the mobile base, and the three-dimensional size of the gripped item. Modeling of the pose uncertainty is collected into the transformation

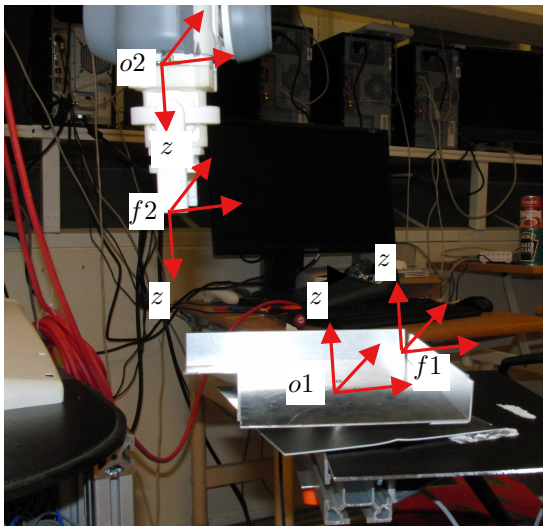

Fig. 3. The frames used to specify the task.

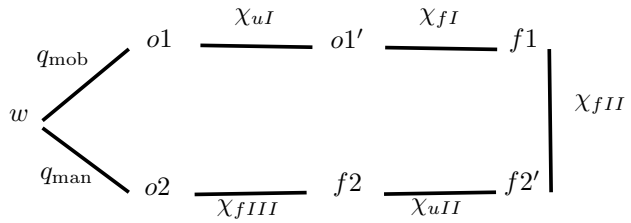

Fig. 4. Feature and uncertainty coordinates and how the frames are connected. The primed frames represent the modeled frames while the others represent the true frames.

between $o1$ and $o1'$ as $\chi_{uI} = (x^{uI} \quad y^{uI} \quad \psi^{uI})^{\mathrm{T}}$. The length, width and height of the gripped item are modeled by the transformation between $f2$ and $f2'$. However, the length, width and height are resolved using force control; that is, the motion in the x, y, and z directions are guarded by a force sensor when searching for contact. When reaching contact in the z-direction the motion is force controlled to maintain contact, while the x and y directions are both position controlled. Therefore no explicit uncertainty coordinates are used to model this uncertainty, which implies that χ_{uII} is of dimension zero. Furthermore, all feature coordinates are chosen as outputs, which implies that the output vector is

$$\begin{aligned} y_1 = x, \quad y_2 = y, \quad y_3 = z, \\ y_4 = \psi, \quad y_5 = \theta, \quad y_6 = \phi. \end{aligned}$$

C. Redundancy Control

The constraints used to specify the task are in this work either position, velocity, or force based. Since the control system for FRIDA only allows to alter the joint velocity and position references, and since there are unmodeled dynamics and disturbances in the system the constraint equation has to include feedback at position, velocity or force level to not violate the constraints in stationarity. Here, a velocity-based control scheme is employed. First generation of the modified desired output velocities, $\dot{y}_d^0$, is done according to

$$\dot{y}_d^0 = f(r, \dot{y}_d, y_m). \tag{1}$$

Here $\dot{y}_d$ denotes the desired output velocities that would be generated if the system would be an ideal velocity controlled system, and r and y_m denotes the reference and measured and/or estimated positions, velocities, or forces. As an example, if an impedance-based position controller is used, (1) will be generated by integrating

$$\ddot{y}_d^0 = \frac{1}{M}(F - D(\dot{y}_d - \dot{x}) - K(y_d - x)) \tag{2}$$

once. In (2), M, D and K are tuning parameters, F is the measured force, and x is the measured and/or estimated position. If force control is used, a force reference is added to the controller in (2). The relation

$$A\dot{q} = \dot{y}_d^0 + B\dot{\chi}_u \tag{3}$$

is used to solve for the joint velocities $\dot{q}$. Here A and B are matrices composed of the different jacobians involved. Output relation (3) is obtained by differentiating the position loop constraints, see [4] for more information.

Since the task specification only has six constraints, three translational and three rotational, and since the mobile base and the manipulator have more degrees of freedom than six the matrix A will not be square. Hence an inverse does not exist. Instead the optimization problem

$$\begin{aligned} \text{minimize} \quad & \dot{q}^{\mathrm{T}} M \dot{q}, \\ \text{subject to} \quad & A\dot{q} = \dot{y}_d + B\dot{\chi}_u \end{aligned} \tag{4}$$

is solved to find a pseudoinverse $A^{\#}$, with the solution

$$A^{\#} = M^{-1}A^{\mathrm{T}}(AM^{-1}A^{\mathrm{T}})^{-1}.$$

When M is chosen as the identity matrix $A^{\#}$ becomes the well known Moore-Penrose pseudoinverse, and (4) then minimizes the norm of the joint velocities.

D. Uncertainty Estimation

The uncertainty coordinates are constituted by the pose of the mobile base— that is, the position in the plane and the yaw angle. By introducing the states as described in Table I, the state vector at time t is given by the 11-dimensional vector

$$x_t = \begin{pmatrix} p_t & v_t & a_t & b_{a,t} & \psi_t & \omega_t & b_{\omega,t} \end{pmatrix}^{\mathrm{T}}.$$

Through modeling the process as a constant acceleration process the model is

$$x_{t+1} = Fx_t + v_t,$$

where

$$F = \begin{pmatrix} I & TI & \frac{T^2}{2}I & 0 & 0 & 0 & 0 \\ 0 & I & TI & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & TI & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{pmatrix}.$$

The submatrices I and 0 are of appropriate dimensions, and

TABLE I

MOBILE BASE STATES WITH DESCRIPTION AND DIMENSION.

State	Description	Dim.
p	Vehicle position in world coordinates	2
v	Vehicle velocity in world coordinates	2
a	Vehicle acceleration in world coordinates	2
b_a	Bias state for acceleration measurement	2
ψ	Yaw angle relative to the world frame	1
ω	Yaw velocity in world coordinates	1
b_ω	Bias state for angular velocity measurement	1

$T = 0.01$ (s) is the sample time. The process noise v_t is assumed independent and Gaussian, according to

$$v_t \sim \mathcal{N}(0, Q),$$
$$Q = GQ^v G^{\mathrm{T}},$$

where

$$G = \begin{pmatrix} \frac{T^3}{6}I & 0 & 0 & 0 \\ \frac{T^2}{2}I & 0 & 0 & 0 \\ TI & 0 & 0 & 0 \\ 0 & TI & 0 & 0 \\ 0 & 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \end{pmatrix},$$

and

$$Q^v = \mathrm{diag}(q_a,\ q_{ba},\ q_\omega,\ q_{b\omega}).$$

All q-variables are process noise variance parameters.

The measurement model is given by

$$z_t = h_t + e_t = \begin{pmatrix} p_t \\ R_w^{o1} v_t \\ R_w^{o1} a_t + b_{a,t} \\ \psi_t \\ \omega_t + b_{\omega,t} \end{pmatrix} + e_t,$$

where R_w^{o1} is the rotation matrix between the mobile base and the world frame origin. The measurement noise e_t is assumed to be independent and Gaussian, leading to

$$e_t \sim \mathcal{N}(0, Q^e),\ Q^e = \mathrm{diag}(q_p,\ q_v,\ q_{ma},\ q_\psi,\ q_{m\omega}).$$

The matrix Q^e is determined experimentally.

Due to different sample rates and computational demands the measurements arrive with different delays, something which causes degraded performance if not accounted for. The described model is used together with an extended Kalman filter (EKF) [21] exploiting the out-of-sequence measurements approach to estimate the pose of the mobile base. See [1] and [2] for algorithm details. Since the vision algorithm has much larger delays than the wheel encoder measurements, only the vision measurements are considered to have any delay.

To summarize the estimation algorithm: When IMU and/or wheel encoder measurements arrive, use the standard EKF to update state estimates and covariances. When a position measurement arrives from the camera, use the out-of-sequence measurements approach to take into account the computational time of the vision algorithm.

IV. PICK-AND-PLACE SCENARIO

The strategy for picking and placing the cans was chosen such that all uncertainties could be eliminated in a robust way while not degrading performance too much. As previously mentioned the plate with borders could simulate a shelf with walls. Therefore it is important that the manipulator's arms do not intersect the extension of the borders. By inspecting the setup in Fig. 2 an intuitive choice would then be to place the item with the right arm (seen from FRIDA), but because of reachability issues and for the purpose of exploiting the extra degree of freedom of the manipulator the left arm was used with the elbow controlled such that collision was avoided. Secondly, the items should be placed one after another. A convenient solution to this was to first search in one direction in the xy-plane. When contact was reached search in the other translational direction was commenced.

A flowchart is shown in Fig. 5. In state 1 the manipulator picked up the item. When entering state 2 the mobile base started moving with varying velocity, and kept moving approximately until the can was released. In state 2 the manipulator moved to a position above the plate, where the orientation was chosen such that frames $f1$ and $f2$ were aligned in the z-direction. In state 3 a search motion using velocity control in the z-direction was made. The position and orientation were kept constant in the other directions using impedance control for the translational movements and PI control for the rotational movements. When contact was made the state machine entered state 4, where the control in the z-direction now switched to an impedance-based force controller. In states 4 and 5 the search for contact using velocity control was done for the y- and x-directions, respectively, which means that the constraints on y_2 and y_1 were velocity constraints. As soon as contact was made for either direction, the controller was switched back to a position-based impedance controller. Finally, the item was released and the procedure could be restarted or terminated.

V. EXPERIMENTAL RESULTS

A. Uncertainty Estimation

For verification of the pose estimation FRIDA was used as ground truth. Given that the control system is ideal the only thing that moves the outputs away from its references are incorrect pose and velocity estimates. In Fig. 6 the results from an experiment is shown where the aim was to keep outputs y_1 and y_2 positioned at the origin when the mobile base drove around. The other outputs were velocity controlled to zero velocity. The upper diagram shows the difference between the mobile base translation estimates and the x- and y-coordinates of the flange, and the lower diagram shows the estimated mobile base velocities.

The minor deviations that occur are when the base accelerates. This is something that can be explained by the relatively low sampling rates of the wheel encoders and vision algorithm, but also because the system is not perfectly controlled. Hence, (1) only keeps the constraints fulfilled at

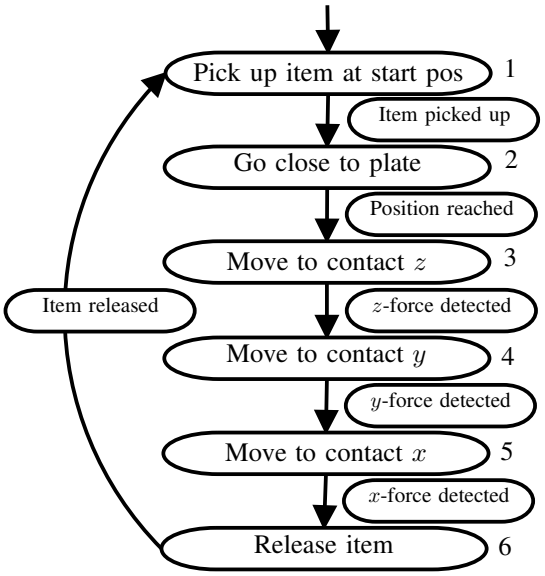

Fig. 5. A flowchart describing the pick and place strategy used in the experiments. From state 2 and forwards the mobile base kept moving approximately until entering state 6.

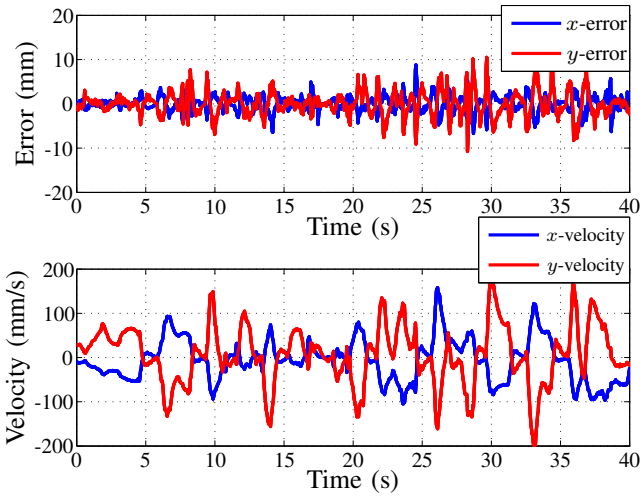

Fig. 6. Verification of the uncertainty estimation algorithm. The upper plot shows the x (blue) and y (red) error between FRIDA's flange and the position of the mobile base when outputs y_1 and y_2 should be kept at zero position. The lower plot shows the estimated velocities of the mobile base, where the x-velocity is shown as blue, and the y-velocity as red. Note that the positions and velocities are shown with respect to the world frame.

stationarity. Note that if the estimation algorithm would not take into account the time delay of the vision algorithm the outputs would violate the constraints also during nonaccelerating movements. The results seem to indicate that the estimation is smooth and correct enough to perform high-precision coordinated control, at least when the acceleration is not too large.

B. Pick-and-Place Scenario

Force data from an experiment is shown in Fig. 7, together with the state sequence. The threshold set to trigger transitions to state 4 is -4 (N) for the z-force, which is seen in Fig. 7. To trigger transitions to states 5 and 6 the threshold is 2 (N) for the y- and x-forces, respectively, which also can be seen in Fig. 7. The y- and x-directions are position controlled using impedance controllers in all states except in states 4 and 5, respectively, where search motions with constant velocity references instead are set. The force in the z-direction is controlled to -5 (N) as soon as contact is made, which it seems to keep reasonably well. The rotational coordinates are velocity controlled to zero velocity in all states from state 3.

Velocity data for the mobile base and desired outputs are shown in Fig. 8 from the same experiment. By inspection the search motions become clear. From approximately $t = 1.8 - 5$ (s) the z-velocity is controlled to keep constant velocity. Between $t = 5 - 8.5$ (s) the y-velocity is controlled, and between $t = 9 - 10.8$ (s) the x-velocity is controlled. The release of the can occurs where the z-control signal goes to nonzero velocity at $t = 10.8$ (s). The path that the mobile base traversed and the state transitions are found in Fig. 9.

VI. CONCLUSIONS AND FUTURE WORKS

A kinematically redundant manipulator was used to perform pick and place on a mobile robot while it was moving. The movement of the mobile robot was estimated with a modified extended Kalman filter. To get increased robustness with respect to the remaining uncertainties force control was used. The constraint-based task specification methodology was exploited to specify the task, taking into account additional sensors, uncertainty estimation and the redundant degrees of freedom. Although the current setup may look quite different from the intended one, they are very similar with respect to the control and kinematics.

Future work includes mounting FRIDA on the mobile base and the force sensor on the wrist. More vision will be used, both for object recognition and for improving the pose estimation algorithm with cameras mounted on the robot for floor-texture detection.

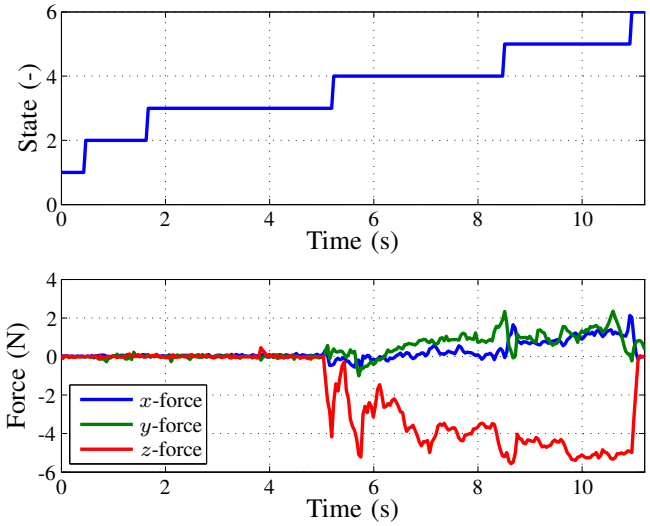

Fig. 7. Force data from an experiment, together with the state information. The x, y, and z forces are shown as blue, green, and red, respectively. The z-force is controlled to -5 (N) from state 4 and onwards. The threshold for activating the force control in the z-direction is set to -4 (N). The output in y- and x-directions are position controlled except in states 4 and 5, respectively, where they are velocity controlled.

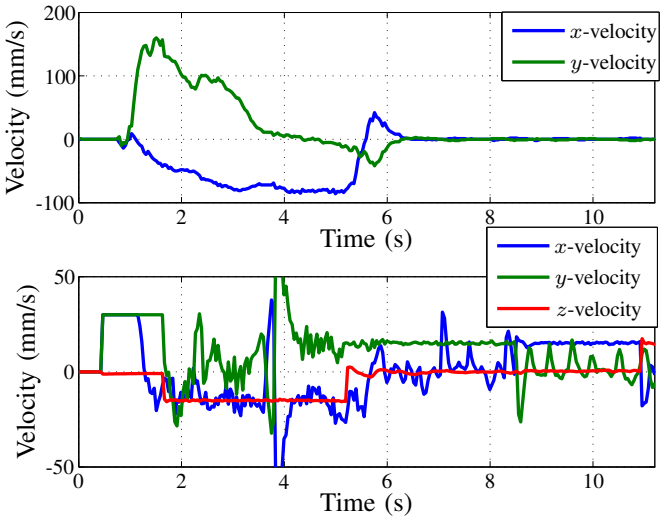

Fig. 8. Velocity data for the mobile base and desired output velocities from an experiment. The upper diagram shows the x-velocity (blue) and the y-velocity (green). The lower diagram shows the x (blue), y (green), and z (red) control signals; that is, the modified desired output velocities in (1). The control signals for the ZYX-Euler angles are not showed, since they were practically zero throughout the experiment. Note that the internal ABB-controller is active in state 1, which is the reason why the control signals show zero velocities although in reality they are not.

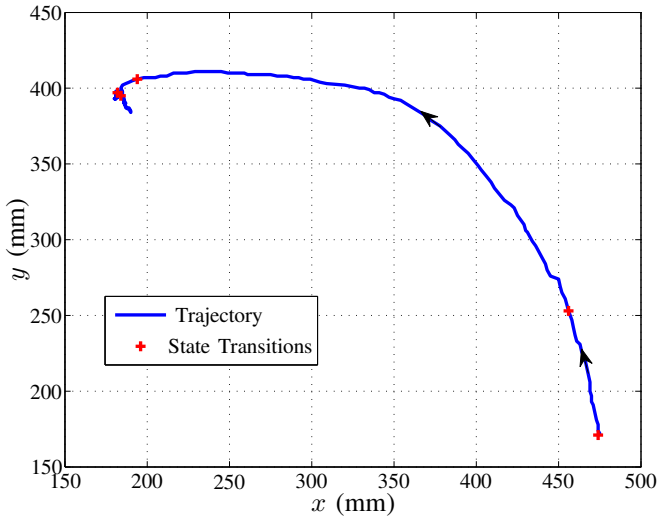

Fig. 9. Traversed path of the mobile base together with the state transitions, shown as red crosses. The path is for the origin of frame $o1'$. The transitions start from state 1 up to state 6, all in all five transitions. The transitions to states 2 and 3 are in the lower right part of the figure, whereas the other three transitions are in the upper left part.

VII. ACKNOWLEDGMENTS

This work was supported by the Swedish Foundation for Strategic Research through the project ENGROSS, by the Swedish Research Council through the LCCC Linnaeus Center, and by the ELLIIT Excellence Center.

REFERENCES

[1] Y. Bar-Shalom, H. Chen, and M. Mallick, "One-step solution for the multistep out-of-sequence-measurement problem in tracking," *Trans. on Aerospace and Electronic Syst.*, vol. 40, pp. 27–37, Jan. 2004.

[2] K. Berntorp, K.-E. Årzén, and A. Robertsson, "Sensor Fusion for Motion Estimation of Mobile Robots with Compensation for Out-of-Sequence Measurements," in *Int. Conf. on Control, Automation, and Systems*, (Seoul, Korea), October 2011.

[3] N. Hogan, "Impedance control: An approach to manipulation," in *American Control Conference*, (San Diego, California, USA), pp. 304–313, June 1984.

[4] J. De Schutter, T. De Laet, J. Rutgeerts, W. Decré, R. Smits, E. Aertbeliën, K. Claes, and H. Bruyninckx, "Constraint-based task specification and estimation for sensor-based robot systems in the presence of geometric uncertainty," *Int. J. Rob. Res.*, vol. 26, pp. 433–455, May 2007.

[5] W. Carriker, P. Khosla, and B. Krogh, "An approach for coordinating mobility and manipulation," in *Proc. Int. Conf. on Systems Engineering*, (Dayton, Ohio, USA), pp. 59–63, Aug 1989.

[6] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation," in *Proc. Int. Conf. on Intelligent Robots and Systems (IROS)*, vol. 2, (Osaka, Japan), pp. 546–553 vol.2, Nov 1996.

[7] O. Khatib, "Mobile manipulation: The robotic assistant.," *Robotics and Autonomous Systems*, pp. 175–183, 1999.

[8] W. Meeussen, M. Wise, S. Glaser, S. Chitta, C. McGann, P. Mihelich, E. Marder-Eppstein, M. Muja, V. Eruhimov, T. Foote, J. Hsu, R. B. Rusu, B. Marthi, G. Bradski, K. Konolige, B. P. Gerkey, and E. Berger, "Autonomous door opening and plugging in with a personal robot," in *Proc. Int. Conf. on Robotics and Automation (ICRA)*, 2010.

[9] L. Petersson and H. I. Christensen, "A framework for mobile manipulation," in *Int. Symp. on Robotics Systems (SIRS)*, (Coimbra, PT), pp. 359–368, July 1999.

[10] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *J. Robotics and Automation*, vol. 3, pp. 43–53, February 1987.

[11] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Trans. Systems, Man and Cybernetics*, vol. 11, pp. 418–432, June 1981.

[12] S. Kock, T. Vittor, B. Matthias, H. Jerregard, M. Kallman, I. Lundberg, R. Mellander, and M. Hedelind, "Robot concept for scalable, flexible assembly automation: A technology study on a harmless dual-armed robot," in *Int. Symp. Assembly and Manufacturing (ISAM)*, pp. 1–5, May 2011.

[13] U. Reiser, R. Klauser, C. Parlitz, and E. Verl, "Desire web 2.0- integration management and distributed software development for service robots," in *Proc. Int. Conf. Adv. Rob. (ICAR)*, (Munich, Germany), 2009.

[14] *MTi and MTx User Manual and Technical Documentation*. Netherlands, 2010.

[15] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, (Osaka, Japan), 2009.

[16] A. Blomdell, G. Bolmsjö, T. Brogårdh, P. Cederberg, M. Isaksson, R. Johansson, M. Haage, K. Nilsson, M. Olsson, T. Olsson, A. Robertsson, and J. Wang, "Extending an industrial robot controller–implementation and applications of a fast open sensor interface," *IEEE Robotics and Automation Magazine*, vol. 12, pp. 85–94, Sept. 2005.

[17] A. Blomdell, I. Dressler, K. Nilsson, and A. Robertsson, "Flexible application development and high-performance motion control based on external sensing and reconfiguration of ABB industrial robot controllers," in *Proc. of the ICRA 2010 Workshop on Innovative Robot Control Architectures for Demanding (Research) Applications*, (Anchorage, AK, USA), June 2010.

[18] D. S. Fors, *Assemblies of Pervasive Services*. PhD thesis, Department of Computer Science, Lund University, 2009.

[19] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE, TRANS. PAMI*, vol. 24, no. 2, pp. 237–267, 2002.

[20] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *Int. J. Rob.*, vol. 26, pp. 519–535, 2007.

[21] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Prentice Hall, 1979.