Mobile Manipulation – Getting a grip?

H. I. Christensen, L. Petersson & M. Eriksson

Centre for Autonomous Systems
Numerical Analysis and Computing Science
Royal Institute of Technology
SE-100 44 Stockholm, Sweden

{hic, larsp, eriksson}@nada.kth.se

Abstract

Mobile Manipulation involves integration of a range of different functionalities from navigation to interaction with objects. To accommodate these functionalities there is a need for a coherent architecture that facilitate such an integration. In addition a rich variety of behaviours with highly variable dynamics must be coordinated. In this paper the basic functionalities for mobile manipulation are discussed. A hybrid deliberative architecture is outlined together with its specification using the hybrid dynamics systems paradigm. Using this framework it is outlined how vision and force torque information may be utilized for recognition, grasping and interaction with objects. Initial experimental results with the integrated system are also presented.

1. Introduction

Recently mobile manipulation has received significant attention [1, 2, 3, 4, 5, 6, 7]. Research has initially been hampered by lack of adequate platforms for such research. Today platforms are available and the question of implementing such systems open up for a range of new research issues. Mobile manipulation is by no means simply a control problem. Overall the mobile manipulation task can be divided into three subproblems: i) localization and servoing on the object to be grasped, ii) (pre-) grasping, and iii) interaction with the object. Each step might at first seem fairly well defined, but once it has to be carried out in a realistic physical setting like a regular home or a factory, a significant number of new issues must be addressed.

Localization and servoing on objects, includes a) recognition of objects and estimation of their pose, b) servoing towards the object of interest, subject to nonholonomic and dynamic constraints, while ensuring stable features that enable sensor based pose-tracking, and c) closed loop control to provide the needed integration of motion of the platform and the manipulator. Other objects in the environment must be taken into consideration to ensure avoidance of these, while grasping the object of interest. Sensing and control must thus satisfy multiple (potentially conflicting) objectives. During this phase the environment is modeled using sensors such as sonars, ir-ranging, laser ranging and computational vision. Vision is particularly important to enable efficient recognition, tracking and pose-estimation. Control of basic mobility can be carried out using one of many well-known techniques reported in the literature, see for example [8, 9]

The grasping phase involves a smooth transition from range/vision driven control to force-torque/tactile control. Vision is not an efficient modality for control once contact has been achieved. Control is here gradually handed over to the force-torque based control, constrained by the available degrees of free motion. This further requires integration of high band-width control with slower systems based on visual feedback, which in turn imposes strong real-time requirements. The area of force based manipulation is extensively described in the literature [10, 11]

During manipulation of objects like doors and drawers the dynamics of the system is highly constrained and there is thus a need for careful planning of the motion taking into account the available degrees of freedom and structures in the environment (i.e. during manipulation the platform must still cope with (dynamic) obstacles). This requires use of efficient methods for real-time (re-)planning and control based on information from multiple sensory modalities [12, 3].

Each of the three problems involved in mobile manipulation are discussed in this paper. Initially an overall architecture for the system is discussed, robust vision for manipulation is then outlined, and methods for control in the context of interaction are discussed. Finally a number of issues related to future research are outlined.

2. System Architecture

Mobile manipulation involves mobile navigation and manipulation. Mobility is typically controlled at a modest sampling rate of 10-25 Hz as the inertia of the system implies limited bandwidth of the closed-loop systems. Manipulation on the other hand involves interaction with objects in the environment and the bandwidth of the system is high, which calls for a system with sampling rates of 500 Hz - 3 kHz. Combining such systems poses a problem as the demands are very different. Standard industrial manipulators have typically 6 degrees-of-freedom. In addition a fully holonomic platform has 3 degrees of freedom. The system thus has 3 extra degrees of freedom for position control of the end-effector, which poses an interesting control problem. Earlier research on redundant manipulation has exploited the null-space / kernel for the manipulator for control of the platform [2], but in general the control can be posed as a constrained optimization problem [13].

Navigation is implemented using a behaviour based approach, following the current trend in mobile systems. The behaviour based system is setup as a hybrid deliberate system, composed of three layers: deliberation (speech input of commands and planning), task management in terms of configuration of groups of behaviors to complete a given task and finally a set of behaviours for coupling of perception to actions. Fusion of different behaviours is accomplished using simple superposition. The manipulation is from a mobility point of view implemented as a behaviour that specifies motion to be carried out to satisfy the manipulation objective. The basic behaviour based system is described in detail in [14].

Manipulation involves a range of different actions for grasping from pre-grasping to full interaction. The control during each phase is different. Pre-grasping involves visual servoing using standard point-to-line image feature alignment. At the final phase of pre-grasping the manipulator is pre-shaped using feed-forward control and the force sensor is used for detection of contact. Upon contact control is driven by force-torque feedback is used to control the manipulator and reference coordinates are forwarded to the platform for maximum agility.

To accommodate these different types of control the manipulator control is implemented using a hybrid control system. In the event of vision guided manipulation the manipulation system is organized as shown in figure 1. The vision based servoing is here part of the general platform system while the hybrid system and joint control is carried out by the higher speed manipulation system.

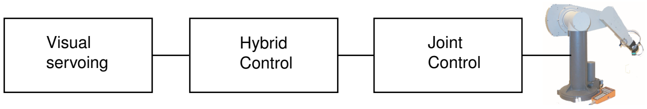

Figure 1: Architecture for control of manipulation system

The hybrid control system is specified by the following 7-tuple:

$$H = (Q, U, I, f, E, G, T)$$

where:

Q is the set of discrete states, typically corresponding to the set of possible tasks.

U is continuous variables, here the joint variables.

I the initial state set, i.e., $I \subset Q \times U$.

f is a vector field that specifies the state dependent control laws, i.e. $Q \times U \rightarrow TU$. Continuity across states is defined in terms of compatibility derived from Luaponov smoothness between control laws.

E is discrete events that specifies state transitions, i.e. events like contact, etc.

G are guards that monitor changes in the continuous variables. When a guard is activated a transition is generated.

T the transition matrix $Q \times \{E \cup G\} \rightarrow Q$, that specifies the relation between states and events.

The hybrid formalism can easily be implemented using a dedicated language that enable specification and it also provides a basis for automated verification/synthesis. Each state in the system is specified by an identifier, a set of discrete events, the transition matrix, and the parameters for the continuous control laws. An example task specification for drawing on a white-board is shown in Figure 2. The full task set for implementation of a simple drawing system is shown in figure 3. The task consists of navigating to the blackboard, approaching the surface (contact is detected by force). The drawing on the board is carried out while maintaining a constant contact force. The actual drawing is composed of several lines and a circle.

```
BEGIN
        NAME = Approach1;
        TYPE = Whiteboard_approach;

        Force = 200; % Threshold when contact
        Filter = 4;  % Select low-pass filter
        MaxX = 700;  % Define out of reach
        Speed = 20;  % Approach with 20mm/s

        TRANSITION[Contact] = Line1;
        TRANSITION[Out_of_Reach] = End;
TION[Finished] = Retract1;
END
```

```
BEGIN
    NAME = Line1;
    TYPE = Whiteboard_line;

    Length = 50;
    Direction = 225;
    Speed = 40;
    Force = 150;
    FGain = 0.00002;

    TRANSI-

END
```

Figure 2: Example tasks specified in the dedicated language for hybrid control

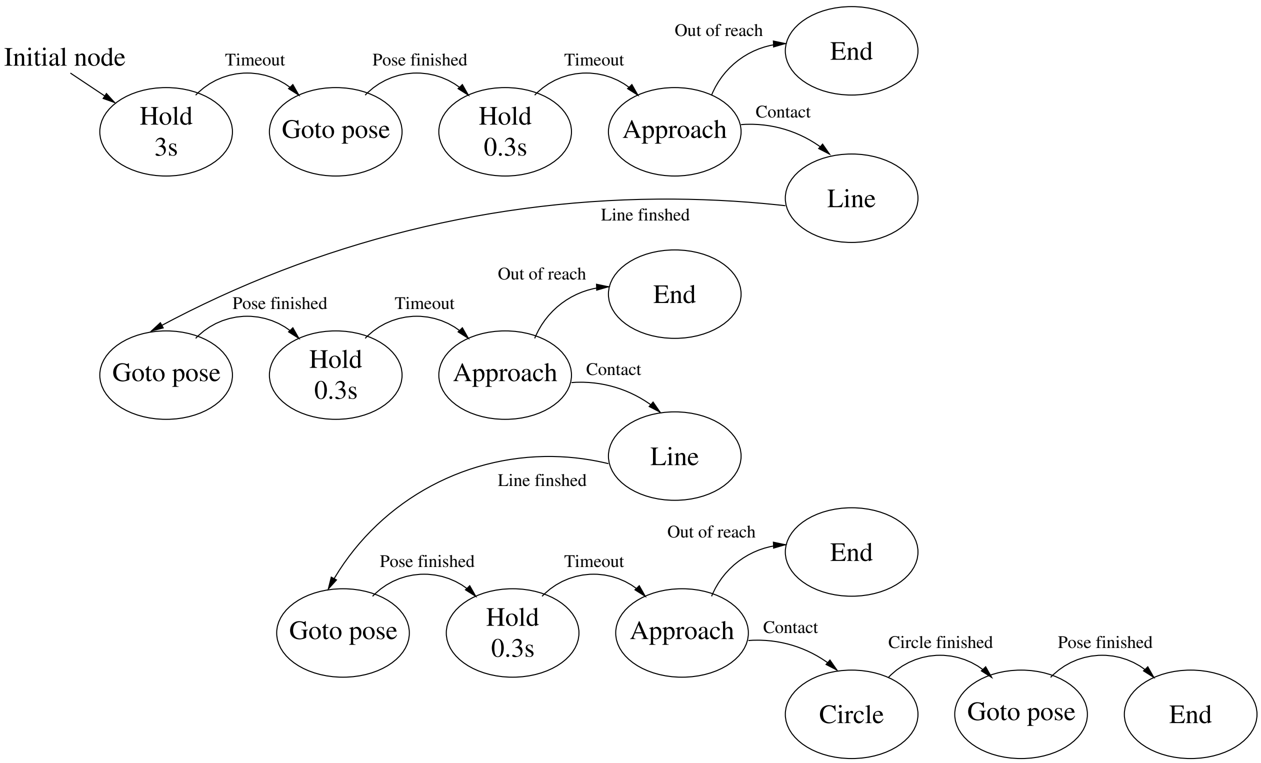

Figure 3: Example of task specification for drawing on a blackboard

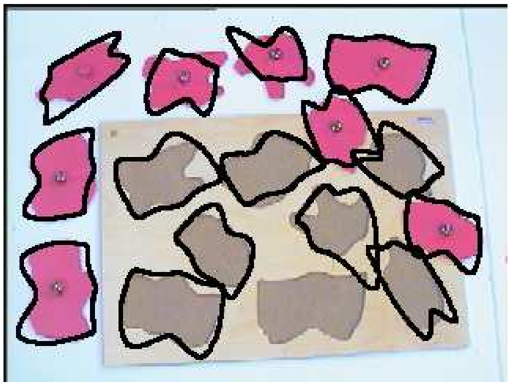

Figure 4: Pieces of a puzzle used for manipulation experiments

3. Vision for Manipulation

In most environments navigation can be achieved using sonars and laser ranging. The limited field of view of these sensors implies that they are unsuitable for control of manipulation. During manipulation there is a need for accurate information about the object to be grasped. This information can be provided by computer vision. Visual invariance has recently gained popularity for 'robust' detection of features and pose estimation for planar objects. A good overview can be found in [15]. In assembly tasks involving objects of limited depth the invariance techniques is a viable alternative. The techniques have been evaluated for automated assembly of puzzles for two year old children. Example pieces are shown in figure 4.

Parameterizing the contours for each piece according to normalized arc-length provides a scale invariant representation where bi-tangents and inflection points can be detected. Both types of features are projectively invariant and can be used for recognition of objects and estimation of pose (here distance and orientation). Detected features are also used for servoing towards the object. The servoing is carried out in image coordinates. The servoing task is in general non-linear, but it can easily be approximated by a first order expansion; i.e.,

$$\dot{X} = J_C^{-1}(X)(F_d - F_i)$$

where J_C is the image Jacobian that maps robot/object motion in Cartesian coordinates into feature variation in image coordinates, while F_d and F_i are the reference position (in the image) and current position, respectively, and $\dot{X}$ is the control vector to the robot. The Cartesian control vector $\dot{X}$ can be converted in the joint control using the inverse kinematics, i.e.:

$$\dot{q} = J_R^{-1}(q)\dot{X}$$

where q are the joint coordinates, and $\dot{q}$ is corresponding velocity control. The closed form solution of J_C for a number of different features combinations has been derived by Hager [16].

The image Jacobian J_C is state dependent and in addition its structure varies in response to the availability of features. This can be used to define a number of different motion strategies as suggested in [16]. This enables definition of a library of motion strategies that can be combined to facilitate on-line selection of control method in response to detected features, i.e. in the event of (temporary) loss of a feature the control strategy can be down-graded to a less optimal strategy. In addition a particular task can be decomposed into a set of different strategies like approach, turn, align, etc. to facilitate more structured design of motion strategies. A detailed description of how this can be accomplished for assembly of the puzzle is provided in [5]

In realistic settings single visual cues are often nonrobust, which results in (temporary) loss of one or more features. Integration of multiple cues, such as colour, texture, motion, edges, etc., enable a significant increase in robustness. One approach is use of voting based techniques [17], that facilitate integration without detailed knowledge of the cue estimators and their interdependencies. Example of use of voting for visual servoing can be found in [18].

4. Force-Torque Based Structure Estimation

Once an object has been detected and servoing has enabled contact, it can be manipulated. In a domestic setting typical tasks include opening of doors and drawers. There are here at least two possible approaches: i) model based and ii) adaptive model estimation. In a model based setting a model for the object of interaction is used for control. Unfortunately it is difficult to provide accurate model. In the second approach a model is estimated only. For the domestic setting linear and rotational motion are considered. The linear motion requires identification of the axis of motion, while the rotational motion (for doors) requires estimation of axis of rotation and radius. The naive approach is force driven interaction, where the motion is driven directly input from the force-torque sensor. The information is rather noisy and it can only be used in low-bandwidth control. An alternative is to estimate model parameters and use it for model based control. In this approach the arm is moved under torque control (low-pass filtered torque data - 64 Hz), through a sequence of steps. Initially the motion is planned to be

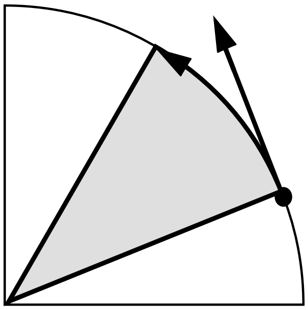

Figure 5: Estimation of axis of rotation and radius from points on a circle

translational and the perpendicular motion (to follow the arc) is determined by a compensating controller (P-Control). A set of points on the arc are used for recursive estimation of the radius and axis of rotation. From basic geometry it is well known that a circle can be written as:

$$r^2 = (x - x_0)^2 + (y - y_0)^2$$

Given three free parameters the system can be solved from three points on the circle arc. Once a set of points have been recovered an initial estimate is available which can be refined recursively. The situation is shown in figure 5. The model based estimation is particularly useful when initial prior information is available about the approximate size of the objects. In the current implementation the objects manipulated are doors where the size is known to be 60-100 cm in radius and the gripper grasps the door by its handle. For this situation the arm is moved at a speed of 20 mm/s along the tangent. The position is compensated at 250 Hz and measurements for the estimation are acquired at 10 Hz. After an initial acquisition of 60 samples (a total motion of 120 mm) the recursive estimation of the radius and center of rotation is initiated (using a cyclic buffer of 60 samples). The approach assumes initially a radius of ∞ and as data points become available a least square fit is performed to determine radius and axis of rotation. A trace of the recursive estimation of radius is shown in figure 6. The reference experiments were carried out with a stick rotating in the horizontal plane to enable easy verification. A source of error in this context could be the horizontal alignment. The method has proved efficient for automatic negotiation of objects as long as the basic type of motion is known (revolute or prismatic).

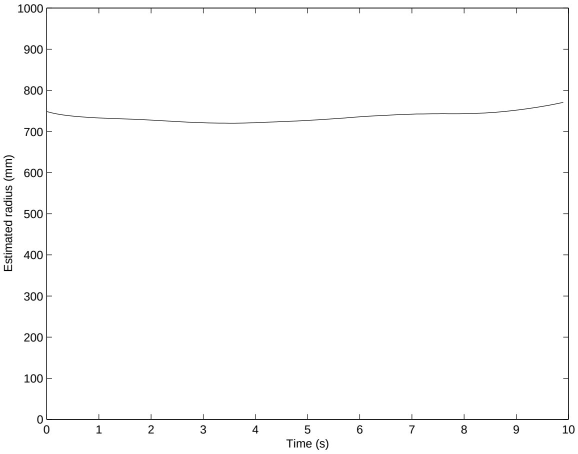

Figure 6: Radius estimation for a 71 cm wide door

5. Implementation

The system outlined above has is implemented on a Nomadic Technologies XR4000 platform with a PUMA 560 mounted on top. The XR4000 platform is fully holonomic with 3 degrees of freedom. The manipulator has 6 degrees of freedom, but the setup with one axis of the manipulator and platform co-incident implies that the combined system has a total of 8 degrees of freedom. The system has an eye-in-hand mounted camera system for visual servoing, a total of 48 ultra-sonic sonars and ir-ranging modules, and a SICK laser scanner for navigation. The arm is further equipped with a JR3 force-torque sensor. Control of the manipulator is carried out by a real-time computer running QNX. The low level control loop runs at 2 kHz while force-based control is achieved at 500 Hz. The behaviour based system is implemented on two separate computers under Linux with the cycle frequency of 10 Hz. The computers are interconnected using TCP/IP implemented on top of a pairwise shared memory system. The platform with the on-board manipulator is shown in figure 7.

6. Summary and Discussion

Mobile manipulation is an emerging field of research. Much early research has focussed on manipulation in the context of well-defined tasks. For operation in a regular house it is, however, difficult to decompose tasks into well defined sub-tasks. Flexibility can be only be achieved through integration of more advanced sensory systems and tighter integration of manipulation and mobility. Control using any sin-

Figure 7: The Nomadic Technologies XR4000 with a PUMA 560 manipulator mounted on top

gle modality is likely to be non-robust and fusion of multiple modalities is thus needed. To facilitate flexible implementation of mobile manipulation a coherent framework for synthesis, analysis and implementation of tasks is needed. One approach to this problem is use of the hybrid dynamic systems (HDS) paradigm. The method provides a framework for explicit analysis which at the same time is well suited for implementation of dedicated control languages that allow easy programming and reconfiguration for different tasks. The HDS approach has been tested in the context of tasks like puzzle solving with success. The major benefit is the easy of transfer of mathematic models into operational systems.

Another problem in mobile manipulation is robust sensory information for navigation, object detection and grasping. A power method for detection and grasping is computer vision. Single visual cues are known to be sensitive to changes in the environment. Through integration of several cues and careful selection of invariant features it is possible to provide reliable features that allow interaction in a natural setting like a home. For general operation in such setting there is a need for flexible perception methods that enable automatic learning of perception strategies for detection and servoing on a rich variety of different objects.

Another problem for manipulation is the need for accurate estimation of the position of the platform that is controlled by a slower system. A consequence of the slower control rate for the platform is that position feedback regarding the platform is unavailable to the control of the manipulator. To enable use of position feedback a state dependent estimation framework is needed.

A recurring problem for mobile manipulation is adaptive handling of variable structure systems. When a mobile manipulator interacts with static structures like doors the control structure of the overall system changes due to imposed non-holonomic constraints. This requires control methods that easily can switch between different control laws while ensuring stability.

Acknowledgments

This research has been sponsored by the Swedish Foundation for Strategic Research through is Centre for Autonomous Systems. The work has benefited significantly from discussion with Prof. Jan-Olof Eklundh, Ms. Danica Kragic, Dr. Magnus Andersson and Mr. Magnus Egerstedt.

References

[1] W. D. Rencken, W. Feiten, and R. Zöllner, *Intelligent Sensor Based Robots*, ch. Relocalisation by Partial Map Matching. Lecture Notes in Artificial Intelligence, Heidelberg: Springer Verlag, July 1999.

[2] O. Khatib, K. Yokoi, K.-S. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Coordination and decentralized cooperation of multiple mobile manipulators," *Journal of Robotic Systems*, vol. 13, pp. 755–764, November 1996.

[3] O. Brock and O. Khatib, "Elastic strips: Real-time path modification for mobile manipulation," in *Eight International Symposium of Robotic Research* (Shirai, ed.), pp. 5–13, Osaka, Japan: Springer Verlag, October 1997.

[4] L. Petersson, M. Egerstedt, and H. Christensen, "A hybrid control architecture for mobile manipulation," in *Proc. of International Robotics Symposium '99 – IROS*, (Korea), IEEE, October 1999. Accepted.

[5] M. Eriksson, H. I. Christensen, and J.-O. Eklundh, "Vision based servoing and grasping with multiple control schemes," in *SIRS '99 – 7th International Symposium on Intelligent Robotic Systems* (H. Araujo and J. Dias, eds.), (Coimbra, PT), pp. 51–60, July 1999.

[6] U. Hanebeck, C. Fischer, and G. Schmidt, "Roman: A mobile robotic assistant for indoor service applications," in *IROS-97*, vol. vol. 2, (Grenoble), pp. 518–525, IEEE, CS Press, June 1997.

[7] W. Feiten, B. Magnussen, G. Hager, and K. Toyama, "Modelling and control for mobile manipulation in everyday environments," in *8th International Symposium of Robotic Research* (Y. Shirai, ed.), pp. 14–22, Osaka, Japan: Springer Verlag, October 1997.

[8] R. C. Arkin, *Behaviour Based Robotics*. Intelligent Robots and Autonomous Agents, Boston, MA: MIT Press, 1998. ISBN 0-262-01165-4.

[9] D. Kortenkamp, P. Bonassa, and R. Murphy, eds., *Artificial Intelligence and Mobile Robots*. Boston, MA: MIT Press/AAAI Press, 1998.

[10] J. D. Schutter and H. V. Brussel, "Compliant robot motion i: A formalism for specifying compliant motion tasks," *International Journal fo Robotics Research*, vol. 7, no. 4, pp. 3–17, 1988.

[11] J. D. Schutter, H. Bruyninckx, W.-H. Zhu, and M. Spong, "Force control: a bird's eye view," in *IEEE CSS/RAS Intl. Workshop on "Control Problems in Robotics and Automation: Future Directions"*, San Diego, CA.: Springer Verlag, 1997.

[12] O. Khatib, K. Yokoi, J. Chang, D. Ruspini, R. Holmberg, A. Casal, and A. Baader, "Force strategies for cooperative tasks in multiple mobile manipulator systems," in *Seventh International Symposium of Robotics Research*, Munich: Springer Verlag, October 1995.

[13] J. Grebäck, "A kinematic approach to motion planning of a mobile manipulator," Master's thesis, Optimization and Systems Theory, Dept of Mathematics, Royal Institute of Technology, Stockholm, Sweden, June 1999. http://www.math.kth.se/ altafini/mobman.

[14] M. Andersson, A. Öreback, M. Lindström, and H. Christensen, *Intelligent Sensor Based Robots*, vol. 1724 of *Lecture Notes in Artificial Intelligence*, ch. ISR: An Intelligent Service Robot, pp. 291–314. Heidelberg: Springer Verlag, October 1999.

[15] J. Mundy and A. Zisserman, eds., *Geometric Invariance for Computer Vision*. Boston, MA: MIT Press, 1992.

[16] G. D. Hager, "A modular system for robust positioning using feedback from stereo vision," *IEEE Transactions on Robotics and Automation*, vol. 13, pp. 582–595, August 1997.

[17] B. Parhami, "Voting algorithms," *IEEE Transactions on Reliability*, vol. 43, no. 3, pp. 617–629, 1994.

[18] D. Kragic and H. Christensen, "Integration of visual cues for active tracking of an end effector," in *Proc. IEEE International Robotics Symposium*, (Seoul), October 1999.