Proceedings of the 39th IEEE
Conference on Decision and Control
Sydney, Australia • December, 2000

Reactive Mobile Manipulation using Dynamic Trajectory Tracking: Design and Implementation[1]

P. Ögren*, L. Petersson†, M. Egerstedt‡ and X. Hu*

*{petter,hu}@math.kth.se
Opt. and Systems Theory
Royal Inst.of Tech.
SE-100 44 Stockholm, Sweden

†larsp@nada.kth.se
NADA/CVAP
Royal Inst.of Tech.
SE-100 44 Stockholm

‡magnuse@hrl.harvard.edu
Div. of Applied Sciences
Harvard University
Cambridge, MA 02138, USA

Abstract

A solution to the trajectory tracking problem for mobile manipulators is proposed and implemented on a real robotic system. Given a trajectory for the gripper to follow, a tracking algorithm for the manipulator is designed, and at the same time the base motions are generated in such a way that the base is coordinated with the gripper while reactively avoiding obstacles. Furthermore, it is shown that the method allows arbitrary upper and lower bounds on the gripper-base distance to be set, and this can be achieved without introducing deadlocks into the system.

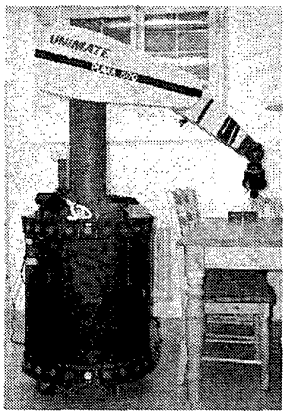

Figure 1: The hardware on which the algorithm was implemented

[1]This work was sponsored in part by the Swedish Foundation for Strategic Research through its Centre for Autonomous Systems at KTH and in part by TFR.

1 Introduction

In this paper, we investigate how to conduct mobile manipulation in such a way that the base motions are influenced by an appropriate, reactive obstacle avoidance behavior [3], together with an arm coordination behavior. At the same time, we want the gripper to track a reference trajectory, evolving in $\mathbb{R}^3$, in such a way that we can impose hard bounds on the gripper-base distance, which corresponds to keeping the gripper in the dextrous workspace [6], relative to the base.

This type of problem is relevant for a number of reasons. For instance, if one wants to be able to orient the gripper arbitrarily, for example when conducting fine object manipulation, then one needs to keep the gripper within this dextrous workspace. Furthermore, when lifting heavy loads, one typically do not want to extend the arm too much, because that could potentially make the platform fall over, or risk component failure due to the large required control torques. Another possible application could be mobile spray painting, where one wants to keep the gripper away from the base, in order to avoid clogging the sensors with paint.

The problem that we will focus on is how to make sure that the end-effector is placed in the appropriate region, while tracking a desired trajectory in a proven stable way. This should be done at the same time as the platform is moving around in an unknown environment, which calls for a safety strategy that basically adds an obstacle avoidance behavior [1] to the base control system. In this paper we thus continue our work in [2], where a trajectory tracking algorithm for mobile manipulators is proposed. Here we show

how our algorithm is implemented on a real robotic system, the XR4000 with a Puma 560 mounted on top, as seen in Figure 1. We also discuss how one has to modify the algorithm in order to implement it.

This paper is organized as follows: In Section 2, we present a control algorithm for the base which is coordinated with the gripper while avoiding obstacles. We then, in Section 3, present a high-level gripper control that is based on a reparameterization of the desired path, using a so called virtual vehicle approach [4], which results in stable trajectory tracking. We furthermore derive some properties about our controllers, such as stability and deadlock-avoidance in Section 4. Finally, in Section 5 and 6 we discuss the hardware and the software of the platform and show some experimental results that indicate that the proposed control strategy works as intended on a real system.

2 Base Control

Our proposed base control is going to be composed of two reactive behaviors [1]: arm coordination and obstacle avoidance.

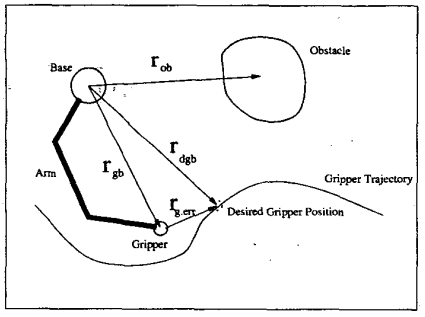

Figure 2: Notation

Before we look into them in detail, we introduce the following notation where all vectors are in an inertially fixed coordinate system (Figure 2) at the same height as the first joint of the robot arm. $\mathbf{r}_b$: vector to base. $\mathbf{r}_g$: vector to gripper. $\mathbf{r}_{dg}$: vector to desired gripper position. $\mathbf{r}_o$: vector to obstacle. $\mathbf{r}_{gb}$: vector to gripper from base. $\mathbf{r}_{ob}$: vector to obstacle from base. $\mathbf{r}_{g.err}$: vector to desired gripper position from gripper. $\mathbf{r}_{dgb}$: vector to desired gripper position from base.

2.1 The Arm Coordination Behavior

Our aim is to make the base maintain an appropriate distance to the gripper, not too far and not too close. We also introduce a dead zone that permits the faster arm to move back and forth without moving the base, as long as the arm is adequately extended.

The singularities of the robot arm (controlling only the spatial position of the wrist) is on the spherical shell where the arm is fully extended and on the vertical axis through the center of the base. Thus the upper bounds on base gripper distance are to be measured from the center of the sphere and the lower bounds as projected distance, i.e. distance from the axis of symmetry.

Introduce the following parameters (all referring to the base-gripper distance) where the first three are cylindrical distances and the last three spherical (as explained above):

- R_{min} – (minimal base-gripper distance)
- R_{LA} – (Lower Acceptable base-gripper distance)
- R_{LG} – (Lower Good base-gripper distance)
- R_{UG} – (Upper Good base-gripper distance)
- R_{UA} – (Upper Acceptable base-gripper distance)
- R_{max} – (maximal base-gripper distance)

The parameters are used in the following way. Outside of (R_{min}, R_{max}) the gripper stops and waits for the base (see gripper control). Inside of (R_{LG}, R_{UG}) the base does not move at all and outside of (R_{LA}, R_{UA}) it moves at full speed (in the absence of obstacles, see below and (Figure 3,4)).

Let $\hat{\mathbf{r}}_{dgb}$ be a normalized projection of $\mathbf{r}_{dgb}$ on the xy-plane.

$$\hat{\mathbf{r}}_{dgb} = \frac{(x_{dgb}, y_{dgb}, 0)}{\sqrt{x_{dgb}^2 + y_{dgb}^2}} \tag{1}$$

Now we have to restrict the height variations on the trajectory to be tracked in order not to have $\mathbf{r}_{dgb} > R_{UG}$ and $\hat{\mathbf{r}}_{dgb} < R_{LG}$ at the same time. We see that if we demand

$$|z_{dg}| \leq \sqrt{R_{UG}^2 - R_{LG}^2}$$

the above can be guaranteed (note that z_{dg} is measured from the height of the first joint of the robot arm).

We propose our arm coordination behavior as

$$\mathbf{v}_{ba} = \begin{cases} -V_{max}\hat{\mathbf{r}}_{dgb} & ||\hat{\mathbf{r}}_{dgb}|| < R_{LA} \\ -V_{max}\frac{R_{LG}-||\hat{\mathbf{r}}_{dgb}||}{R_{LG}-R_{LA}}\hat{\mathbf{r}}_{dgb} & ||\hat{\mathbf{r}}_{dgb}|| \in [R_{LA}, R_{LG}] \\ 0 & \text{(else)} \\ V_{max}\frac{||\mathbf{r}_{dgb}||-R_{UG}}{R_{UA}-R_{UG}}\hat{\mathbf{r}}_{dgb} & ||\mathbf{r}_{dgb}|| \in [R_{UG}, R_{UA}] \\ V_{max}\hat{\mathbf{r}}_{dgb} & R_{UA} < ||\mathbf{r}_{dgb}|| \end{cases} \quad (2)$$

as illustrated in Figure 3.

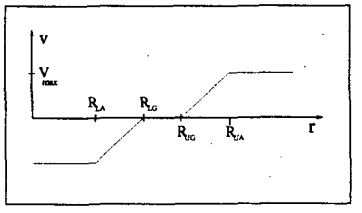

Figure 3: The control gain of the arm coordination behavior as a function of the distance between the base and the desired gripper position.

2.2 The Obstacle Avoidance Behavior

In this paper we assume that the obstacles are fairly low so they can only block the the base, but not the gripper which is mounted on the top of the base. These obstacles can thus be detected in a straight forward manner since since most available robotic platforms, including our Nomad XR4000 (see Figure 1) are equipped with obstacle sensors (sonars and laser scanners).

In order to make the base move smoothly between the obstacles we want the obstacle avoidance behavior to be given by a repulsive vector field, generated from the sensor data.

A natural way of choosing the vector fields for small obstacles is to give them spherical symmetry. With this constructions we can also prove the absence of deadlocks (see Theorem 4.3). We let the norm of the vectors be zero outside of R_U, we increase it linearly to V_{max} at R_L, and we let it be constant, V_{max}, inside of R_L. The direction of the vectors are radial outwards from the obstacle.

$$\mathbf{r}_{ob} = \mathbf{r}_o - \mathbf{r}_b \quad (3)$$

$$\hat{\mathbf{r}}_{ob} = \frac{\mathbf{r}_{ob}}{||\mathbf{r}_{ob}||} \quad (4)$$

$$\mathbf{v}_{bo} = \begin{cases} -V_{max}\hat{\mathbf{r}}_{ob} & ||\mathbf{r}_{ob}|| < R_L \\ -V_{max}\frac{R_U-||\mathbf{r}_{ob}||}{R_U-R_L}\hat{\mathbf{r}}_{ob} & R_L < ||\mathbf{r}_{ob}|| < R_U \\ 0 & R_U < ||\mathbf{r}_{ob}|| \end{cases}$$

This way of giving the vector fields a spherical symmetry is of course not reasonable for large objects such as walls. For them we suggest a vector field pointing outwards from the closest point on the obstacle with norm decreasing in distance as above, i.e. constant V_{max} while inside of some R_L, then decreasing to zero at some other R_U.

In order to avoid local minima we want every possible robot position to be influenced by at most one obstacle.

2.3 The Arbitration

We now apply an arbitration mechanism similar to the one suggested by Arkin [3].

$$\mathbf{v}_b = \frac{\mathbf{v}_{ba} + \mathbf{v}_{bo}}{2} \quad (5)$$

This alone gives some unstable stationary points. In order to avoid these we add the following to our control scheme. If $\mathbf{v}_{ba} = -\mathbf{v}_{bo} \neq 0, ||\mathbf{v}_{ba}|| = V_{max}$ then we let:

$$\mathbf{v}_b = \epsilon : \epsilon \perp \mathbf{v}_{ba}, ||\epsilon|| < V_{max} \quad (6)$$

Since we have the same upper limit on both the behavior gains we have the following.

Theorem 2.1 *If* $||\mathbf{r}_{ob}|| \leq R_L$ *then* $d/dt(||\mathbf{r}_{ob}||) \geq 0$.

Proof. $||\mathbf{r}_{ob}|| \leq R_L$ which implies $||\mathbf{v}_{bo}|| = V_{max}$ and thus $d/dt(||\mathbf{r}_{ob}||) = -\mathbf{v}_b^T\mathbf{r}_{ob}/||\mathbf{r}_{ob}|| = (-\cos(\alpha)||\mathbf{v}_{ba}|| + V_{max})/2 \geq 0$ since $||\mathbf{v}_{ba}|| \leq V_{max}$. □

Thus within R_L of an obstacle the base will not move closer to it independently of the arm coordination behavior. In theorem 4.3 we show that we are still safe from deadlocks under certain assumptions.

3 The Gripper Control

Throughout the paper we make use of the fact that the arm kinematics can be decoupled from the system

[5]. Let $\mathbf{r}_{dg}$ be the reference point on the trajectory that we are tracking. As in [4] we now let the motion of the point, along the trajectory, be governed by a differential equation that contains the tracking error. We furthermore augment this equation with a factor $C(\mathbf{r}_{dgb})$ that depends on the base position. This makes the combined controllers robust and it also allows us to be able to guarantee some properties that we want our system to exhibit, as will be seen in the next section.

$$\mathbf{r}_{dg} = \mathbf{r}_{dg}(s(t)) \quad (7)$$

$$\dot{\mathbf{r}}_{dg} = \mathbf{r}'_{dg}\dot{s} \quad (8)$$

$$\mathbf{r}_{g.err} = \mathbf{r}_{dg} - \mathbf{r}_g \quad (9)$$

$$\dot{s} = \frac{1}{||\mathbf{r}'_{dg}||} v_0 e^{-||\mathbf{r}_{g.err}||} C(\mathbf{r}_{dgb}) \quad (10)$$

$$\dot{\mathbf{r}}_{dg} = \mathbf{r}'_{dg} \frac{1}{||\mathbf{r}'_{dg}||} v_0 e^{-||\mathbf{r}_{g.err}||} C(\mathbf{r}_{dgb}) \quad (11)$$

The coordination factor (see Figure 4) is given by:

$$C(\mathbf{r}_{dgb}) = \quad (12)$$

$$\begin{cases} 0 & ||\hat{\mathbf{r}}_{dgb}|| < R_{min} \\ \left(1 - \frac{R_{LG} - ||\mathbf{r}_{dgb}||}{R_{LG} - R_{min}}\right) & R_{min} < ||\hat{\mathbf{r}}_{dgb}|| < R_{LG} \\ 1 & \text{(else)} \\ \frac{R_{max} - ||\mathbf{r}_{dgb}||}{R_{max} - R_{UG}} & R_{UG} < ||\mathbf{r}_{dgb}|| < R_{max} \\ 0 & R_{max} < ||\mathbf{r}_{dgb}|| \end{cases}$$

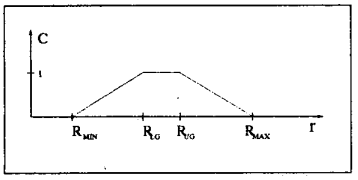

Figure 4: The coordination factor $C(\mathbf{r}_{dgb})$

Thus we stop the desired gripper position $\mathbf{r}_{dg}$ if $||\mathbf{r}_{dgb}|| \notin (R_{min}, R_{max})$. It remains stationary until the base has caught up i.e. $||\mathbf{r}_{dgb}|| \in (R_{min}, R_{max})$. To make the gripper track $\mathbf{r}_{dg}$ a simple proportional controller is proposed.

$$\mathbf{v}_g = \gamma \mathbf{r}_{g.err} \quad (13)$$

Where γ can be any positive constant.

4 Performance Analysis

We will show the following: The base will avoid obstacles and get within (R_{min}, R_{max}) of $\mathbf{r}_{dgb}$ in finite time, (Theorem 4.1 and 4.2). The system will not stop in a deadlock, (Theorem 4.3). The upper and lower bounds on the gripper-base distance can be arbitrarily set, (Theorem 4.4). The non-transient gripper error can be made arbitrarily small, (Theorem 4.5). These properties together show that the suggested control scheme should perform well.

The first property assures that the base moves efficiently around the obstacles. The second property shows that we are always making progress and stops in the desired gripper position are only momentary. The third property assures that the workspace of the arm is bounded. Thus we can ensure that the gripper control does not try to move the gripper outside of its hardware limits. We note that not over extending the arm is also important if we are carrying a heavy load, extending the arm too far can cause tipping or other failures due to the large torques. Finally the fourth property verifies that the trajectory tracking works.

Since this paper focuses on the implementation we refer to [2] for the proofs.

Theorem 4.1 *If the trajectory is feasible and* $||\mathbf{r}_{dgb}(t)|| > R_{max}$ *then there exists a finite* $T > 0$, *such that*

$$||\mathbf{r}_{dgb}(t+T)|| < R_{max}$$

Theorem 4.2 *If the trajectory is feasible and* $||\mathbf{r}_{dgb}(t)|| < R_{min}$ *then there exists a finite* $T > 0$ *such that*

$$||\mathbf{r}_{dgb}(t+T)|| > R_{min}$$

Theorem 4.3 *If the trajectory is feasible there will never be a deadlock, i.e.* $\forall t_0$ *such that* $\dot{s}(t_0) = 0\ \exists t_1 > t_0 : \dot{s}(t_1) \neq 0$.

Theorem 4.4 *The upper and lower bounds on the base-gripper distance can be set arbitrarily.*

$$R_{min} - ||\mathbf{r}_{g.err}|| \leq ||\mathbf{r}_{gb}|| \leq R_{max} + ||\mathbf{r}_{g.err}|| \quad (14)$$

Remark. Note that this bounds the real base gripper distance since $||\mathbf{r}_{g.err}||$ is bounded by Theorem 4.5. The right choice of (R_{min}, R_{max}) can thus guarantee that the gripper is always in the reachable or dextrous workspace.

Theorem 4.5 *While in the reachable workspace, the non-transient gripper error can be made arbitrarily small by adjusting γ:*

$$||\mathbf{r}_{g.err}|| < ||e^{-\gamma t}\mathbf{r}_{g.err}(0)|| + (1 - e^{-\gamma t})/\gamma \quad (15)$$

5 Implementation

The algorithm described above was implemented on a real robot which is shown in Fig. 1. The experimental platform was manufactured by Nomadic Technologies and was of the type XR4000 which is a holonomic platform with three degrees of freedom, translation in the plane and rotation. On top of the platform is an industrial manipulator mounted, a Puma560, to add manipulation skills to the system. The manipulator has six degrees of freedom, which means that the overall system becomes redundant. This can be used as an advantage in that it is possible to compensate for smaller misplacements of the base with a correction of the manipulator pose.

To control the complete system two computers are used. A high and accurate sampling rate is needed because of the very fast dynamics of the manipulator arm, hence a real-time operating system is necessary for the control. In this system, the real-time operating system QNX was chosen for this purpose. An important note to make is that the base has very different dynamic properties, it is much slower than the manipulator and it is therefore possible to control it using the non real-time OS Linux.

Also, a third computer exists in this system but it is not being used in this particular experiment. It is normally dedicated to computationally heavy tasks like computer vision.

A subset of the software used in the intelligent service agent project [7, 8] was used as a framework when implementing the algorithm described in this article. The software consists of several modules controlling different parts of the system. In particular, a separation is made between the platform control system and the manipulator control system. All the software is written in regular C++ which makes it possible to give the robot architecture a good structure.

A set of software PID-controllers are used to do the low-level control of the joints of the manipulator. These were set to run at 1000 Hz and in the following experiments they were receiving control commands at 65 Hz. The presented algorithm generates Cartesian motion commands and therefore it was necessary to use inverse- and forward kinematics to calculate the joint angles. The Puma560 has a symbolic solution to the inverse kinematics which makes it efficient to calculate.

Due to the slow dynamics of the base, a rate of as low as 2 Hz was used to control the translational motion of the platform. However, the sampling rate of the odometry was approximately 5 Hz. This sampling rate of the odometry is enough for slow tracking speeds, but has to be higher when faster and more accurate tasks are performed.

The experimental data presented in the results section was collected at the rate of control, i.e. 65 Hz.

6 Experimental Results

In order to implement the control scheme on a real platform there is some changes that need to be done. The equation (10) must first be discretized. This is done the standard way with time step given by the update frequency of 65Hz. Secondly the arm controller only admits position control. Therefore equation (13) is modified into

$$\mathbf{r}_{g.out} = (1 - \gamma)\mathbf{r}_{gb} + \gamma\mathbf{r}_{dgb}$$

where the new γ is in $(0, 1)$ which as equation (13) yields an exponential motion towards $\mathbf{r}_{dgb}$. After measuring the robot the constants were set to $R_{min} = 0.4m$, $R_{LA} = 0.48m$, $R_{LG} = 0.56m$, $R_{UG} = 0.64m$, $R_{UA} = 0.72m$, $R_{max} = 0.8m$. And the constants $V_{max} = 0.2m/s, \gamma = 0.05$.

The motion was somewhat jerky. This is mainly due to the fact that the odometry only runs at 5Hz. Thus every 0.2s the gripper is faced with a step in the position error that has to be corrected. This hypothesis was supported by the fact that a fourier transform of the data in fig 7 shows a major peak at 5Hz.

In Figure 5 the gripper curve is a sinusoidal passing above the obstacle. The two circles are the inner and outer radii of the obstacle avoidance field with $R_L = 0.3m, R_U = 0.6m$. The trajectory of the actual gripper is too close to the gripper curve to be distinguished at this scale. The trajectory of the base can be seen to avoid the inner obstacle circle. Points on the gripper curve and base curve are connected by straight lines indicating the concurrent positions of gripper and base at one time instant.

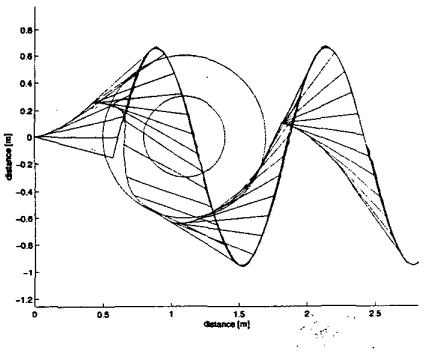

Figure 5: Experimental results, note the difference between the base trajectory in the left (avoiding obstacle) and right part of the plot.

In Figure 6 the distance between the desired gripper position and the base position is displayed as a function of the x coordinate of the desired gripper position. Notice how this distance is in the dead zone at the second peak of the sinus curve. Here the base stands still while the gripper tracks the curve at a convenient distance from the base. At about 2.3m it is even a bit below the dead zone and the base reverses just a bit as can be seen in the very small kink of the base curve in Figure 5. Then the distance increases again as the curve moves away from the base and just before 2.5m the base starts to move again.

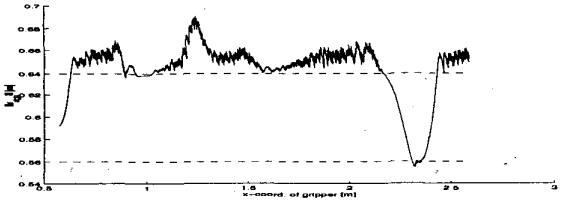

Figure 6: Distance between the desired gripper position and the base position

Figure 7 depicts the distance between the gripper position and the virtual gripper position. Since the idea of the tracking is that the gripper should 'follow' the virtual gripper along the curve, this distance can not be identically or approaching zero (as shown in theorem 4.5). The fact that the axis scales differ by a factor close to 200 makes the oscillations look bigger.

7 Conclusions

In this paper, we show by theory and experimental results how mobile manipulation can be performed in a systematic, and stable way, at the same time as we allow for the base to be influenced by reactive, safety behaviors. Experimental results show that our control approach does not only work in theory but also in practice.

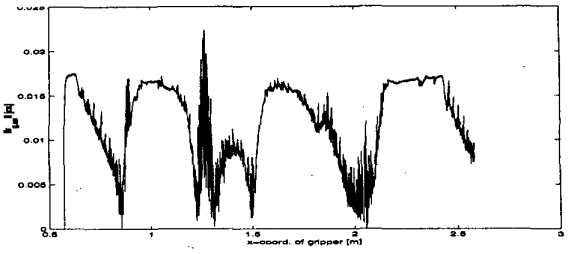

Figure 7: The distance between $\mathbf{r}_{gb}$ and $\mathbf{r}_{dgb}$. Notice how this distance is lowest when the base stands still while the gripper moves in the dead zone. The peak at 1.25 is when the base is most influenced by the obstacle.

References

[1] R.C. Arkin and D.C. Mackenzie. Planning to Behave: A Hybrid Deliberative/Reactive Robot Control Architecture for Mobile Manipulation. *ISRAM '94*, Maui, Hawaii, 1994.

[2] P. Ogren, M. Egerstedt, and X. Hu. Reactive Mobile Manipulation using Dynamic Trajectory Tracking. *IEEE Conference on Robotics and Automation*, San Francisco, California, April. 2000

[3] R.C. Arkin. *Behavior-Based Robotics*. The MIT Press, Cambridge, Massachusetts, 1998.

[4] M. Egerstedt, X. Hu, and A. Stotsky. Control of a Car-Like Robot Using a Virtual Vehicle Approach. *37th IEEE Conference on Decision and Control*, Tampa, Florida, Dec. 1998.

[5] M. Egerstedt, and X. Hu. Coordinated Trajectory Following for Mobile Manipulation *IEEE Conference on Robotics and Automation*, San Francisco, California, April. 2000

[6] R.M. Murray, Z. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[7] M. Andersson and A. Oreback and M. Lindstrom and H.I. Christensen. : Intelligent Sensor Based Robot Systems *Lecture Notes in Artificial Intelligence*. Springer Verlag, Heidelberg, Germany, September 1999.

[8] L. Petersson, M. Egerstedt, and H.I. Christensen: A Hybrid Control Architecture for Mobile Manipulation. *1999 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Kyongju, Korea.