Coupling Mobile Base and End-Effector Motion in Task Space

Tim Welschehold[1] Christian Dornhege[1] Fabian Paus[2] Tamim Asfour[2] Wolfram Burgard[1]

Abstract— **Dynamic systems are a practical alternative to motion planning in executing robot actions. They are of particular interest in Learning from Demonstration, as here we aim to carry out actions in a certain fashion, which might be difficult to achieve with a planner. Using model-based dynamic systems in task space enables robots to flexibly reproduce demonstrated actions. Nevertheless, when dealing with mobile manipulators, we face the challenge of including the kinematic constraints of the robot in the action models. In this paper we propose to couple robot base and end-effector motions generated by arbitrary dynamical systems modulating the base velocity, while respecting the robots kinematic design. To this end we learn an approximation of the inverse reachability in closed form. In real-world robot experiments we demonstrate that we are able to maintain kinematically feasible trajectories in the presence of obstacles and in configurations differing profoundly from the training scene.**

I. INTRODUCTION

The rising expectations on mobile manipulators to serve as service robots in custom households in the near future calls for fast development of versatile deployable task learning methods. Using dynamical systems for trajectory generation in robotics has the advantage that one can encode complex robot behavior in a flexible fashion using motion primitives [1] or statistical models [2] in task space. In contrast to planning approaches [3], where a specific trajectory for an action execution is hard to implement, motion patterns can easily be learned. In our previous work we presented a method to learn mobile manipulation actions from human demonstrations [4], [5]. The approach records data of human trajectories and adapts these to the robots capabilities including feasible grasp and kinematics. From the adapted data we learn motion models in task space using dynamic systems for trajectory generation. While we ensure that the data used for learning is kinematically sound, no such guarantee can be given for the motion generated by the learned model. This is especially relevant in the presence of obstacles not present in the trained scenes or if performing the learned actions from starting configurations differing considerably from the seen demonstrations. It is still a hard problem in general to generate combined trajectories for base and end-effector from motion models. While planning approaches circumvent this difficulty implicitly by exploring paths in the robot configuration space, this is not easy to cope when using dynamical systems operating in task space.
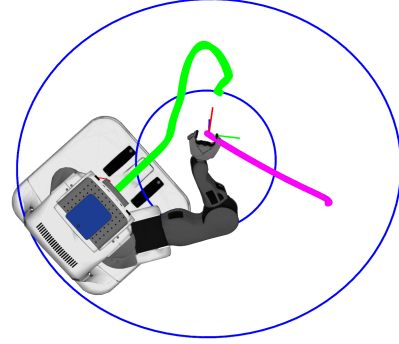
Fig. 1: Illustration of the envisioned coupling of base and manipulator motion. Due to the robots kinematics, for a given end-effector pose, the possible locations for the base are restricted to the area bounded by the two blue ellipses, which approximate the inverse reachability. Trajectories for the PR2's base and end-effector are displayed in green and magenta. The ellipses are plotted for the displayed end-effector pose.

In this work we propose an approach that treats the inverse reachability constraint as an obstacle avoidance problem for the robot base motion. The following contributions are made in this paper. First we present an extension to real-time obstacle avoidance in dynamical systems [6] to couple formerly independent robot base and end-effector motion. Furthermore we present a method to approximate inverse reachability by geometric objects to allow real-time usage. This representation is learned from inverse reachability maps (IRM) [7]. In real-world robot experiments we demonstrate the feasibility of our approach on a mobile robotic platforms, the PR2. We show that we are able to generate kinematically sound trajectories performing a variety of manipulation actions.

II. RELATED WORK

Dynamical systems are a popular method for flexible motion generation in the field of Learning from Demonstration. Pastor *et al.* [1] rely on dynamic movement primitives to learn a parametrized non-linear differential equation system to reproduce demonstrated motion. Calinon *et al.* [2] follow a similar procedure by estimating the dynamical system's parameters in a Gaussian mixture model. In this work we use action learning procedures building upon these methods. Khansari-Zadeh *et al.* [6] propose an approach for dynamic obstacle avoidance applicable with system like the above mentioned. Parts of our method are inspired by their system. Inverse reachability maps (IRM) have been introduced in [7] and [8]. Here we use them as foundation for a formulation usable in dynamic systems. A different approach to achieve kinematically sound robot behavior is

pursued by Stulp *et al.* [9]. They learn to position the robot through trial-and-error interaction with the environment. No joint base and end-effector motion is considered though. Most work on mobile manipulation approaches the problem from a planning perspective and acts in robot configuration space while including some task space constraints [3], [10], [11]. Mike *et al.* [12] present an experience-graph planning approach seeded with kinesthetic demonstrations. Constraints in task space addressing similar actions as carried out by us are used. Again, no joint motion of base and end-effector is considered. Another approach building on planned paths in joint space was presented in the elastic strips framework by Brock *et al.* [13]. Obstacles are included as potential fields and the resulting forces are mapped to joint displacements using a kinematic model of the manipulator. There is some work on planing joint base and end-effector motion in task space. Leidner *et al.* [14] implement a similar approach for robot positioning as [7] and further extend it to plan sparse Cartesian trajectories with respect to the object for a given task whilst performing reachability checks.

III. APPROACH

Given some dynamic system of the form $\dot{\boldsymbol{\xi}} = f(\boldsymbol{\xi})$ to generate trajectories for base and end-effector of a mobile manipulator we introduce a concept to couple the two motions in a kinematically sound way. To this end we translate the platforms inverse reachability constraints to an obstacle avoidance problem depending on the relative pose of base an end-effector. All handled poses $\boldsymbol{\xi}$ and their corresponding velocities $\dot{\boldsymbol{\xi}}$ are assumed to be given in state space. Given an initial state $\boldsymbol{\xi}_0$ a motion trajectory can be gained by integrating the system recursively. Our approach on coupling base and end-effector motion is based on the obstacle avoidance method proposed by Khansari-Zadeh *et al.* [6]. In the following we give a brief summary of their approach followed by our proposed extension to account for the inverse reachability.

A. Dynamical System Approach to Obstacle Avoidance

A convex obstacle centered at the origin can be described by a continuous function $\Gamma(\boldsymbol{\xi}) : \mathbb{R}^d \mapsto \mathbb{R}$. The function Γ is C^1 smooth and increases monotonically in each dimension with distance to the center. On the surface of the object $\Gamma(\boldsymbol{\xi})$ equals 1, in the free region around the obstacle we have $\Gamma(\boldsymbol{\xi}) > 1$. For each point $\boldsymbol{\xi}$ outside the obstacle Khansari-Zadeh *et al.* define a deflection hyperplane given by its normal $n(\tilde{\boldsymbol{\xi}})$:

$$n(\boldsymbol{\xi}) = \left[\frac{\partial \Gamma(\boldsymbol{\xi})}{\partial \xi_1} \cdots \frac{\partial \Gamma(\boldsymbol{\xi})}{\partial \xi_d} \right]^T \qquad (1)$$

They further define a basis for the deflection hyper-plane

$$e_j^i(\boldsymbol{\xi}) = \begin{cases} -\dfrac{\partial \Gamma(\boldsymbol{\xi})}{\partial \xi_i} & j = 1 \\[2mm] \dfrac{\partial \Gamma(\boldsymbol{\xi})}{\partial \xi_1} & j = i \neq 1 \\[2mm] 0 & j \neq 1, j \neq i \end{cases} \quad i \in 1..d-1, j \in 1..d \qquad (2)$$

Given some dynamics $f(\boldsymbol{\xi})$, which generates a velocity $\dot{\boldsymbol{\xi}}$, they formulate a modulation of said velocity as:

$$\dot{\boldsymbol{\xi}} = \bar{M}(\boldsymbol{\xi}) f(\boldsymbol{\xi}) \qquad (3)$$

Where $\bar{M}(\tilde{\boldsymbol{\xi}})$ is a modulation matrix describing the effect of K obstacles at poses $\boldsymbol{\xi}_0^k$ in the scene

$$\bar{M}(\boldsymbol{\xi}) = \prod_{k=1}^{K} M^k(\tilde{\boldsymbol{\xi}}^k) \qquad (4)$$

where $\tilde{\boldsymbol{\xi}}^k = \boldsymbol{\xi}_0^k \otimes \boldsymbol{\xi}$ describes the pose relative to obstacle k. Each modulation matrix $M^k(\tilde{\boldsymbol{\xi}}^k)$ describes a transformation into a basis E, followed by a scaling in its i composing directions by a factor λ^i and a transformation back to the world reference system. In the following the subscript k is omitted on M, E, D, n and λ^i for simplicity. Each $M^k(\tilde{\boldsymbol{\xi}}^k)$ is assembled as

$$M(\tilde{\boldsymbol{\xi}}) = E(\tilde{\boldsymbol{\xi}}) D(\tilde{\boldsymbol{\xi}}) E(\tilde{\boldsymbol{\xi}})^{-1} \qquad (5)$$

where the matrix E is an orthonormal basis

$$E(\tilde{\boldsymbol{\xi}}) = \begin{bmatrix} n(\tilde{\boldsymbol{\xi}}) & e^1(\tilde{\boldsymbol{\xi}}) & \dots & e^{d-1}(\tilde{\boldsymbol{\xi}}) \end{bmatrix} \qquad (6)$$

and D a diagonal matrix

$$D(\tilde{\boldsymbol{\xi}}) = \begin{bmatrix} \lambda^1(\tilde{\boldsymbol{\xi}}) & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \lambda^d(\tilde{\boldsymbol{\xi}}) \end{bmatrix} \qquad (7)$$

the λ^i for each obstacle k are calculated as

$$\lambda^i(\tilde{\boldsymbol{\xi}}^k) = \begin{cases} 1 - \dfrac{\omega^k(\tilde{\boldsymbol{\xi}}^k)}{|\Gamma(\tilde{\boldsymbol{\xi}}^k)|}, & i = 1 \\[3mm] 1 + \dfrac{\omega^k(\tilde{\boldsymbol{\xi}}^k)}{|\Gamma(\tilde{\boldsymbol{\xi}}^k)|}, & 2 \leq i \leq d \\[3mm] 1, & \end{cases} \qquad \begin{aligned} & \\ & n(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}_{rel}^k < 0 \\ & \\ & n(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}_{rel}^k \geq 0 \end{aligned} \qquad (8)$$

For $i = 1$, which corresponds to the part parallel to the surface normal, this slows down the velocity while increasing it in the other directions to circuit the obstacle, see Fig. 2. $n(\boldsymbol{\xi})^T \dot{\boldsymbol{\xi}} \geq 0$ means moving in direction of the obstacle surface normal, i.e., away from the object, in which case the velocity should not be modulated.

The weights $\omega^k(\tilde{\boldsymbol{\xi}}^k)$ for each obstacle depend on the distances from $\tilde{\boldsymbol{\xi}}$ to all surfaces ($\Gamma(\tilde{\boldsymbol{\xi}}) = 1$) and are designed to be 1 if on the surface of obstacle k and 0 if on another obstacles surface.

$$\omega^k(\tilde{\boldsymbol{\xi}}^k) = \prod_{i=1, i \neq k}^{K} \frac{\Gamma^i(\tilde{\boldsymbol{\xi}}^i) - 1}{(\Gamma^i(\tilde{\boldsymbol{\xi}}^i) - 1) + (\Gamma^k(\tilde{\boldsymbol{\xi}}^k) - 1)} \qquad (9)$$

Consequently $\lambda^1 = 0$ on the border of object k and we have

$$n(\tilde{\boldsymbol{\xi}})^T \dot{\xi}_{rel}^k = 0 \qquad (10)$$

on the border of the object, guarantying the impenetrability of the surface [6]. For an in-depth description of this approach we refer the reader to [6].
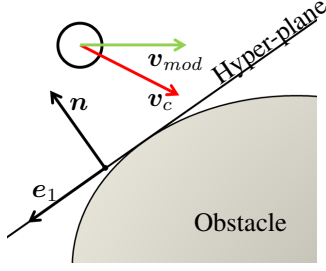
Fig. 2: For a convex obstacle, we define a deflection hyper-plane orthogonal to the surface. The robots velocity $\boldsymbol{v}_c$ is accelerated along the line basis, i.e., $\boldsymbol{e}_1$ and decelerated orthogonally to it, resulting in a modulated velocity $\boldsymbol{v}_{mod}$.

B. Coupling Base and Gripper

For our approach we do not only intend for the robots base to avoid obstacles but also remain in a bounded region given by the end-effector pose and its inverse reachability. We assume that we have a continuous region of reachability bounded by one outer and one inner surface with the same properties as required for the obstacles above, see Fig. 4. The definition of the boundaries for our setting will be discussed in Sec. III-D. Assuming we have an adequate description of these boundaries, in the following we discuss necessary adjustments to couple robot base and gripper motion.

We have an inner boundary due to kinematic constraints of the robot that do not allow the base to get arbitrary close to the end-effector. This kinematically infeasible region is treated similarly to ordinary obstacles. We introduce a factor $a < 1$ in $\lambda(\tilde{\boldsymbol{\xi}}^k)$ for the scaling of the escape motion parallel to the surface to avoid unwanted orbiting of the base around the end-effector.

$$
\lambda^i(\tilde{\boldsymbol{\xi}}^k) = \begin{cases} 1 - \dfrac{\omega^k(\tilde{\boldsymbol{\xi}}^k)}{|\Gamma(\tilde{\boldsymbol{\xi}}^k)|}, & i = 1 \quad \wedge \, \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} < 0 \\ 1 + \dfrac{a \cdot \omega^k(\tilde{\boldsymbol{\xi}}^k)}{|\Gamma(\tilde{\boldsymbol{\xi}}^k)|}, & 2 \leq i \leq d \wedge \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} < 0 \\ 1, & \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} \geq 0 \end{cases}
\tag{11}
$$

Analogously we have a bound for the maximum distance the base can have from the gripper. For this outer boundary we want the base to stay inside the bounded region instead of outside. To keep a consistent notation we simply invert $\Gamma'(\tilde{\boldsymbol{\xi}}) = \frac{1}{\Gamma(\tilde{\boldsymbol{\xi}})}$ in a first additional step. Further we only aim at keeping the robot inside the bound and not circling it so that there is no need in modulating the velocity in directions other than the surface normal. Accordingly we set $\lambda(\tilde{\boldsymbol{\xi}}^k)$:

$$
\lambda^i(\tilde{\boldsymbol{\xi}}^k) = \begin{cases} 1 - \dfrac{\omega^k(\tilde{\boldsymbol{\xi}}^k)}{|\Gamma'(\tilde{\boldsymbol{\xi}}^k)|}, & i = 1 \quad \wedge \quad \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} > 0 \\ 1, & 2 \leq i \leq d \quad \vee \quad \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} \leq 0 \end{cases}
\tag{12}
$$

The modulation proposed by Khansari-Zadeh *et al.* acts on the relative velocity between the robot and the obstacles. They only consider static obstacles, thus all having same velocities. It is not straight forward to account for varying

velocities among different obstacles and the impenetrability constraint Eq. (10) might be violated. To overcome this issue we propose to define a weighted mean relative velocity $\dot{\tilde{\boldsymbol{\xi}}}$ based on $\omega^k(\tilde{\boldsymbol{\xi}}^k)$

$$
\dot{\tilde{\boldsymbol{\xi}}}_{base} = \dot{\boldsymbol{\xi}}_{base} - \frac{1}{\sum \omega^k} \sum_{k=0}^{K} \omega^k(\tilde{\boldsymbol{\xi}}^k) \cdot \dot{\boldsymbol{\xi}}^k
\tag{13}
$$

This averaged velocity is modulated and afterwards transformed back to the stationary world system:

$$
\dot{\boldsymbol{\xi}}'_{base} = \bar{M}(\boldsymbol{\xi}) \cdot \dot{\tilde{\boldsymbol{\xi}}}_{base} + \frac{1}{\sum \omega^k} \sum_{k=0}^{K} \omega^k(\tilde{\boldsymbol{\xi}}^k) \cdot \dot{\boldsymbol{\xi}}^k
\tag{14}
$$

In the proximity of one obstacle this averaged relative velocity will converge towards the relative velocity regarding said obstacle and thus ensuring the impermeability of its surface. When simultaneously approaching two obstacles with different velocities this cannot be insured. When only dealing with static obstacles and moving inverse reachability surfaces according to the end-effector motion, this is solved by slowing the gripper motion $\dot{\boldsymbol{\xi}}'_{gripper}$ with respect to the $\Gamma(\tilde{\boldsymbol{\xi}}^k)$ value of the nearest obstacle (or inverse reachability surface) k:

$$
\dot{\boldsymbol{\xi}}'_{gripper} = \dot{\boldsymbol{\xi}}_{gripper} \cdot (\Gamma(\tilde{\boldsymbol{\xi}}^k) - 1), \quad \text{if } \Gamma(\tilde{\boldsymbol{\xi}}^k) < c,
\tag{15}
$$

for some $c > 1$. This way the impenetrability constraint Eq. (10) can be adhered for continuous time evolution.

C. Discrete Time-Step Adjustment

Due to the nature of the application in practice we do not have a continuous time flow but rather small integration times to compute the evolution of the system. Therefore the impermeability of the object boundaries cannot be guarantied at all times [6]. We propose to introduce a negative λ^1 for the surface normal when already in collision and velocity towards center of object. This is of special relevance for the soft bounds of the kinematic constraints since we here we need to consider two, potentially opposing, motions as well as changing shape of the bounds, see Sec. III-D.

$$
\lambda^1(\tilde{\boldsymbol{\xi}}^k) = \begin{cases} -b, & \Gamma(\tilde{\boldsymbol{\xi}}^k) < 1 \, \wedge \, \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} > 0 \\ b, & \Gamma(\tilde{\boldsymbol{\xi}}^k) < 1 \, \wedge \, \boldsymbol{n}(\tilde{\boldsymbol{\xi}})^T \dot{\boldsymbol{\xi}}^k_{rel} < 0 \end{cases}
\tag{16}
$$

With $b \gg 1$ the velocity component in direction of the hyperplane normal will be increased thus avoiding deeper penetration of the bounds and drive the robot out of the obstacle on a short path.

D. Modeling Inverse Reachability as Obstacles

We now aim to model the inner and outer boundaries of the kinematic constraints for the two degrees of freedom in the translation of the base motion as obstacles based on the inverse reachability given by the IRM. To gain continuous regions, we choose geometric primitives, in our case ellipses as they match the IRM boundaries of the robot well (see Fig. 4). The shape of these depends on the gripper configuration, i.e., the height above the ground z_g

and its pitch and roll angles β_g, γ_g. The end-effector position (x_g, y_g) and yaw θ_g have no influence on the shape as the ellipse can simply be transformed to that pose. For many manipulators, including the PR2's, the dependence on the roll angle γ_g can also be omitted due to the 360° freedom of the wrist roll joint. Therefore, we determine a mapping from the height z_g and pitch β_g to ellipsoids with parameters $(c_x, c_y, a_x, a_y, \alpha)$, defining each ellipses center, major and minor axis and rotation angle. In a first step we create N_z discretizations of z_g and N_β discretizations of β_g into a total of K tuples (z_g^k, β_g^k) of intersections of the IRM with the ground each providing sets of points (X^k, Y^k) that represent valid base positions relative to the end-effector. For each of these K sets we extract the inner and outermost points, which are then used to fit K ellipses $(c_x^k, c_y^k, a_x^k, a_y^k, \alpha^k)$ (see Fig. 3 left). To this end, we define an objective function

$$\Psi = \sum_{k=1}^{K} e(X^k, Y^k) + \omega_1 \sum_{j=2}^{N_z} e(z_{j-1}, z_j) + \omega_2 \sum_{l=2}^{N_\beta} e(\beta_{l-1}, \beta_l) \tag{17}$$

that considers the error for each of the K ellipses to the respective K IRMs by summing the distances of all points in the set (X^k, Y^k) to the ellipse borders, i.e.,

$$
\begin{aligned}
e(X^k, Y^k) = \\
\sum_{i}^{I} \Bigg(1 - \left(\frac{(x_i - c_x)\cos(\alpha) + (y_i - c_y)\sin(\alpha)}{a_x} \right)^2 + \\
\left(\frac{(y_i - c_y)\cos(\alpha) - (x_i - c_x)\sin(\alpha)}{a_y} \right)^2 \Bigg)^2 .
\end{aligned}
\tag{18}
$$

It additionally minimizes differences between neighboring ellipses by the deviation in parameters depending on z or β weighted with ω_i.

$$e(z_m, z_n) = \sum_{i=1}^{N_\beta} \| \boldsymbol{c}_{\beta_i}^{z_m} - \boldsymbol{c}_{\beta_i}^{z_n} \|^2 + \| \boldsymbol{a}_{\beta_i}^{z_m} - \boldsymbol{a}_{\beta_i}^{z_n} \|^2 + d(\alpha_{\beta_i}^{z_m}, \alpha_{\beta_i}^{z_n})$$

$$e(\beta_m, \beta_n) = \sum_{i=1}^{N_z} \| \boldsymbol{c}_{z_i}^{\beta_m} - \boldsymbol{c}_{z_i}^{\beta_n} \|^2 + \| \boldsymbol{a}_{z_i}^{\beta_m} - \boldsymbol{a}_{z_i}^{\beta_n} \|^2 + d(\alpha_{z_i}^{\beta_m}, \alpha_{z_i}^{\beta_n})$$

$$\tag{19}$$

with $d(\alpha_m, \alpha_n)$ as angular distance. By minimizing Ψ we obtain sets of ellipse parameters, both for the inner and outer bound, for each of the k tuples of (z_g^k, β_g^k).

In the next step we generate a continuous mapping $(z_g, \beta_g) \mapsto (\boldsymbol{c}, \boldsymbol{a}, \alpha)$ for the inner and outer ellipses. To this end, we use the estimated values from above to build a Gaussian process model for each of the ellipse parameters, see Fig. 3. We use a squared exponential covariance function for $\boldsymbol{a}$ and $\boldsymbol{c}$ and a periodic one for α. We perform hyperparameter optimization with Rprob [15]. During trajectory generation we use Gaussian process regression to retrieve the parameters for the ellipses for the evolving gripper configuration (z_g, β_g). Finally, we use the center and axis of the ellipse to transform the robot base pose to the respective ellipse's frame as $\tilde{\boldsymbol{\xi}}$ to gain the boundary as an ellipse
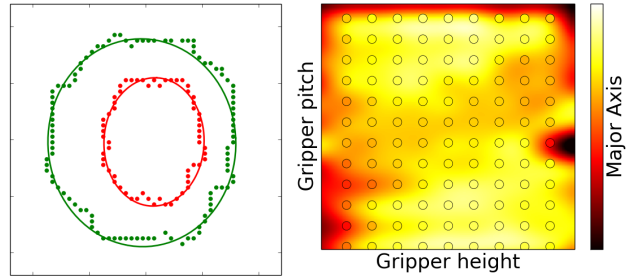

Fig. 3: The left image shows fits for the inner (red) and outer (green) ellipses generated for a gripper pose of $(z_g = 0.9, \beta_g = -0.261)$. The green and red points mark the outer and inner bounds of the considered IRM intersection with the ground. The right image shows the prediction of a Gaussian Process for the outer ellipses major axis. x and y axis correspond to z_g and β_g in ranges of 0.3-1.3 m and $-\pi/2$-$\pi/2$ radian respectively, the color intensity illustrates predicted values in a range of 0.1-1 m. The training points are superimposed and marked as circles.
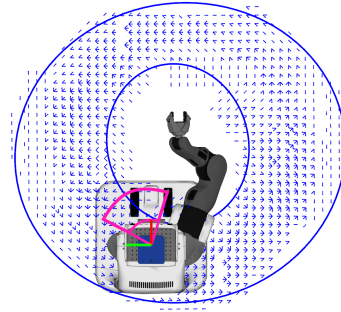

Fig. 4: Intersection of the IRM of the PR2 with the ground for the illustrated robot pose. Legal base poses are marked as small blue arrows. The ellipses represent our proposed approximation. The magenta circle sector at the base pose corresponds to an interval for sound base orientations.

function $\Gamma(\tilde{\boldsymbol{\xi}})$ given as

$$\Gamma(\tilde{\boldsymbol{\xi}}) = \left(\frac{\tilde{\boldsymbol{\xi}}_x^{base}}{a_x} \right)^2 + \left(\frac{\tilde{\boldsymbol{\xi}}_y^{base}}{a_y} \right)^2 . \tag{20}$$

E. Constraining the Base Orientation

So far we only discussed how to adapt the translational motion of the robot base and not its rotation. Since we are dealing with omnidirectional wheeled platforms our base orientation is 1-dimensional with current value θ_c and turning velocity ω. Due to kinematic limitations the orientation of the robot θ_c cannot be arbitrary. We model this as a region of possible orientations given by a center θ_0 and an aperture θ_b, see Fig. 4.

To estimate θ_0 and θ_b we generate a lookup table for (θ_0, θ_b) from the inverse reachability map for the base position by locally averaging for θ_0 and determining the local bounds for θ_b. Depending on the robot configuration, i.e., the gripper height z_e and pitch β_e and the base position (x_b, y_b) we retrieve θ_0, θ_b and update them on trajectory generation via k-nearest neighbor regression for the evolving robot configuration. Since θ_0 is a circular quantity we calculate its mean over corresponding points in the unit circle for the k nearest neighbors. As the lookup parameters (z_e, β_e, x_b, y_b)

correspond to different physical entities we include a scaling for the nearest neighbor lookup. The pitch angles are scaled such that $1° \mathrel{\widehat{=}} 2\,\text{cm}$. Once we have (θ_0, θ_b) we define a function Γ_{θ_c} indicating acceptability of θ_c

$$\Gamma_{\theta_c}(\theta_c, \theta_0, \theta_b) = \left(\frac{\theta_c - \theta_0}{\theta_b} \right)^p \qquad (21)$$

for some even $p \geq 2$. Depending on $\Gamma_{\theta_c}(\theta_c, \theta_0, \theta_b)$ we define a modulation factor λ_{θ_c}

$$\lambda_{\theta_c} = \begin{cases} 1, & \text{if } \Gamma_{\theta_c} < 1 \ \wedge \ (\theta_c - \theta_0) \cdot \omega \geq 0 \\ (1 - \Gamma_{\theta_c}), & \text{if } \Gamma_{\theta_c} < 1 \ \wedge \ (\theta_c - \theta_0) \cdot \omega < 0 \\ \Gamma_{\theta_c}, & \text{if } \Gamma_{\theta_c} \geq 1 \ \wedge \ (\theta_c - \theta_0) \cdot \omega \geq 0 \\ -\Gamma_{\theta_c}, & \text{if } \Gamma_{\theta_c} \geq 1 \ \wedge \ (\theta_c - \theta_0) \cdot \omega < 0 \end{cases} \qquad (22)$$

and adapt the angular velocity ω of the robots base.

$$\omega' = \omega \cdot \lambda_{\theta_c} \qquad (23)$$

In the first two cases with $\Gamma_{\theta_c} < 1$ the orientation is within the bounds and is either left unchanged if velocity is pointing to the region center, i.e., $(\theta_c - \theta_0) \cdot \omega \geq 0$, or slowed otherwise. If the base orientation is in an undesirable configuration, i.e., $\Gamma_{\theta_c} \geq 1$, we accelerate to the center θ_0.

IV. EVALUATION AND EXPERIMENTS

In this section we evaluate the introduced methods and show their applicability in real-world robot experiments.

A. Evaluation of Reachability Approximation

First we compare the learned approximation for the inverse reachability with the original IRM. To evaluate the approximation we determine the false discovery rate, i.e., poses that are not reachable according to the IRM but are within our approximation bounds, divided by all poses inside our approximation. Besides this, the miss rate, i.e., poses that are legal with respect to the IRM but are not contained in our approximation, divided by the total number of poses within the IRM is relevant. We consider the inverse reachability for gripper heights $z \in (0.4, 1.2)m$ and gripper pitch angles $\beta \in (-\pi/2, \pi/2)$ radian. The evaluation is performed on grids with a step size of $5\,\text{cm}$ in the IRM of the corresponding (z, β) layer. We achieve an overall false discovery rate of 2.8% and a miss rate of 10.7%. This shows that the risk of falsely accepting critical poses is low. The slightly higher miss rate is not problematic since it only means that we potentially discard reachable poses. This indicates that the ellipses indeed are a good approximation of the IRM shape. The Gaussian process regression of the ellipses parameters is smooth with z_g and β_g, compare right image in Fig. 3 for the example of the major axis of the outer ellipse. The results are similarly smooth for the other ellipse parameters.

B. Evaluation of Generated Trajectories

Next we test the joint robot base and end-effector motion generation using the approximated inverse reachability as the foundation for their coupling. In [4], [5] we described how to learn actions from human demonstration and how they can

	Avg change to demo	Kinematically successful poses		#runs		total #runs
		w/	w/o	w/	w/o	
Grasp handle	0.35 m, 0°	100 %	96.4 %	4	2	4
Grasp handle	0.58 m, 100°	99.8 %	90.6 %	3	0	4
Grasp handle	0.55 m, 270°	100 %	98.8 %	4	3	4
Grasp handle	0.98 m, 175°	99.5 %	90.0 %	3	0	4
Open door	obstacle influence	100 %	96.8 %	1	0	1
Grasp cabinet	no obstacle	100 %	94.7 %	8	3	8
Grasp cabinet	additional obstacle	100 %	65.4 %	8	0	8
Open cabinet	no obstacle	100 %	100 %	1	1	1
Open cabinet	additional obstacle	99.1 %	51.8 %	3	1	4

TABLE I: Evaluation of the generated trajectories. The averaged distances refer to the translational and rotational deviations for the base pose in training and execution. w/ and w/o refer to evaluation applying our approach respectively without it. The given percentage of successful poses is a mean over all runs for the setting. The last two columns specify the number of successful runs without kinematic failures.

be imitated using dynamical systems. Here we evaluate our approach for some of the actions learned in [4], [5], namely grasping a room door handle, opening and driving through the door, grasping a cabinet door handle and opening it. For the evaluation we computed trajectories for the mentioned tasks, using the system described in [4], with and without the proposed modulation based on inverse reachability. We compare the percentage of poses with no kinematic solution for the generated base and end-effector trajectories in Table I for different settings. For the two grasping task we vary the robot's starting configuration towards higher distances as seen in the demonstrations in the learning phase. For grasping the door handle we divided the trials into four groups depending on the starting configurations, see Table I. The given distance to the demonstrations is a mean over the trials. For grasping the cabinet door handle we also included a wide range of initial poses and added an obstacle in the scene. Trials with and without obstacles from Table I were performed with identical starting poses. For the task of opening the room and cabinet doors the starting pose is highly constrained due to the grasped handles. For the room door however, the presence of the door frame as obstacle has a high influence on the dynamic trajectory generation. For opening the cabinet door we added obstacles at different locations influencing the generation of the base trajectory. As Table I shows, our approach is able to flexibly react to strong deviations in starting configuration as well as changes in the environment. Of the 38 trials only 3 contained kinematically critical poses and even in these cases the amount of failures was below 1%. Generating trajectories without applying our proposed modulation only works for cases with small deviations from training and little obstacle influence. Examples for two of the trials are shown in Fig. 5.

C. Validation of Execution of Trajectories

In a last step we validate the generated trajectories by executing them on the PR2 both in simulation and real world experiments for the actions mentioned above. We execute each action five times both in simulation and real world. For the grasping actions we vary the initial poses while the opening actions are executed starting from the end
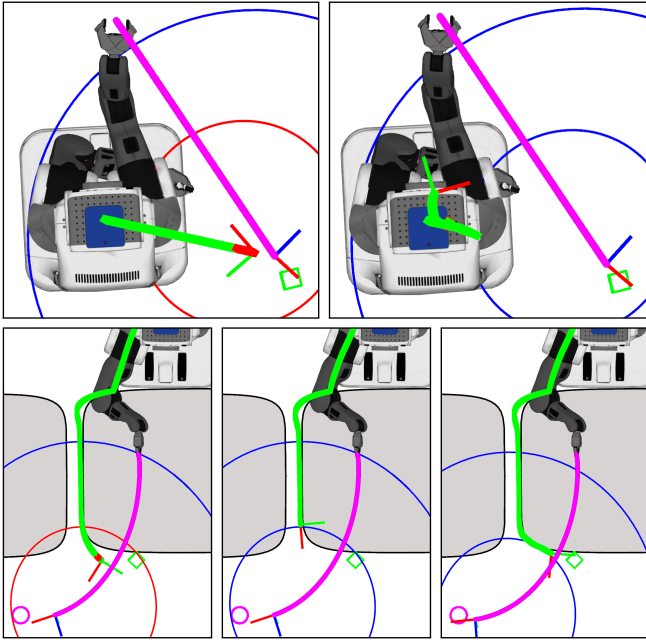
Fig. 5: Exemplary work cases of the proposed approach with the PR2 robot. Generated trajectories are displayed in magenta for the gripper, green and red for kinematically feasible and infeasible base poses. The green square and magenta circle indicate the goal poses. In the top row the task is to grasp a door handle which is to right of the robot (not in image). On the top left we generated end-effector and base trajectories without considering the inverse reachability. We see that the base trajectory penetrates the inverse reachability bounds, causing impracticable configurations. On the top right the base trajectory respects the constrains imposed by the inverse reachability ellipses resulting in a different,executable, trajectory. The bottom row shows the task of opening and driving through a door. The obstacles representing the door frame are shown as the gray area. Again on the left we encounter kinematic problems which can be avoided by using our proposed approach like depicted in the center, and right image, at same execution time respectively shortly after the left image.

configurations of the respective grasping actions. We observe that all trajectories without kinematically infeasible poses could be executed without problems. Even a small amount of kinematically critical poses were handled by the controllers of the robot. At these poses the end-effector motion stops until the next kinematically sound pose is reached resulting in small jumps on execution. Long segments of unreachable poses, however, resulted in inexecutable trajectories. These cases were only observed without our proposed modulation.

V. CONCLUSION AND OUTLOOK

We presented a method to approximate the inverse reachability of mobile platforms by a parametrized ellipse based on inverse reachability maps. We further introduced an approach to use this representation to couple independent robot base and end-effector motion with regard to kinematic feasibility. In our evaluation we show that our system is able to ensure kinematic feasibility when generating trajectories for mobile manipulation actions with dynamic systems. In robot experiments we demonstrate that we can generalize our action models regarding varying robot starting configurations and adapt to changes in the scene. In future work we aim at

Fig. 6: The PR2 executing generated trajectories starting from a pose as seen on the left image. The cabinet, highlighted by the red box, is grasped (top right) and opened (right bottom).

developing a free approximation of the inverse reachability bounds without restriction to a certain geometric form. Depending on the nature of the task, an inversion of the reactivity could be beneficial, where the end-effector velocity is modulated to evade the base motion. Further, a combined approach to modulate the base position and orientation could be addressed instead of having independent corrections.

REFERENCES

[1] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Int. Conf. on Robotics & Automation (ICRA)*, 2009.

[2] S. Calinon, Z. Li, T. Alizadeh, N. G. Tsagarakis, and D. G. Caldwell, "Statistical dynamical systems for skills acquisition in humanoids," in *Int. Conf. on Humanoid Robots (Humanoids)*, 2012.

[3] M. Stilman, "Global manipulation planning in robot joint space with task constraints," *IEEE Transactions on Robotics*, vol. 26, pp. 576–584, 2010.

[4] T. Welschehold, C. Dornhege, and W. Burgard, "Learning mobile manipulation actions from human demonstrations," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[5] C. Zimmermann, T. Welschehold, C. Dornhege, T. Brox, and W. Burgard, "3d human pose estimation in rgbd images for robotic task learnin," in *Int. Conf. on Robotics & Automation (ICRA)*, 2018.

[6] S. M. Khansari-Zadeh and A. Billard, "A dynamical system approach to realtime obstacle avoidance," *Auton. Robots*, vol. 32, no. 4, pp. 433–454, 2012.

[7] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Robot placement based on reachability inversion," in *Int. Conf. on Robotics & Automation (ICRA)*, 2013.

[8] F. Zacharias, C. Borst, and G. Hirzinger, "Capturing robot workspace structure: representing robot capabilities," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2007, pp. 3229–3236.

[9] F. Stulp, A. Fedrizzi, and M. Beetz, "Learning and performing place-based mobile manipulation," in *2009 IEEE 8th International Conference on Development and Learning*, June 2009, pp. 1–7.

[10] F. Burget, M. Bennewitz, and W. Burgard, "Bi2rrt*: An efficient sampling-based path planning framework for task-constrained mobile manipulation," in *Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016.

[11] B. Cohen, S. Chitta, and M. Likhachev, "Search-based planning for dual-arm manipulation with upright orientation constraints," in *Int. Conf. on Robotics & Automation (ICRA)*, 2012.

[12] P. Mike, V. Hwang, S. Chitta, and M. Likhachev, "Learning to plan for constrained manipulation from demonstrations," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, 2013.

[13] O. Brock and O. Khatib, "Elastic strips: A framework for motion generation in human environments," *The International Journal of Robotics Research*, vol. 21, no. 12, pp. 1031–1052, 2002.

[14] D. Leidner, A. Dietrich, F. Schmidt, C. Borst, and A. Albu-Schäffer, "Object-centered hybrid reasoning for whole-body mobile manipulation," in *Int. Conf. on Robotics & Automation (ICRA)*, 2014.

[15] M. Blum and M. Riedmiller, "Optimization of Gaussian Process Hyperparameters using Rprop," in *Eur. Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2013.