2009 IEEE International Conference on Robotics and Automation
Kobe International Conference Center
Kobe, Japan, May 12-17, 2009

Biologically-inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance

Heiko Hoffmann, Peter Pastor, Dae-Hyung Park, and Stefan Schaal

Abstract— **Dynamical systems can generate movement trajectories that are robust against perturbations. This article presents an improved modification of the original dynamic movement primitive (DMP) framework by Ijspeert et al [1], [2]. The new equations can generalize movements to new targets without singularities and large accelerations. Furthermore, the new equations can represent a movement in 3D task space without depending on the choice of coordinate system (invariance under invertible affine transformations). Our modified DMP is motivated from biological data (spinal-cord stimulation in frogs) and human behavioral experiments. We further extend the formalism to obstacle avoidance by exploiting the robustness against perturbations: an additional term is added to the differential equations to make the robot steer around an obstacle. This additional term empirically describes human obstacle avoidance. We demonstrate the feasibility of our approach using the Sarcos Slave robot arm: after learning a single placing movement, the robot placed a cup between two arbitrarily given positions and avoided approaching obstacles.**

I. INTRODUCTION

Humanoid robot assistants have great potential in the future for helping people who cannot help themselves. Likewise, robotic prostheses will be of great benefit for those who have partly or completely lost a limb. Currently, dexterous humanoid robots (e.g., Sarcos CB [3]) and robotic prosthetic arms (e.g., Dean Kamen's Luke Arm [4]) are available. But we lack the knowledge of controlling them efficiently. For example, still many robots try to follow precomputed trajectories. Such an approach, however, is unsuitable in a human environment, which presents many unexpected changes. For prosthetic robot arms, another dominant control approach is to continuously update the position of the end-effector in close-loop through the human; for the control command, for example, a joystick in the shoes [4] or recording from the motor cortex [5] have been explored. Such continuous control, however, presents a high cognitive and attentional load for the human operator.

Dynamical systems offer an alternative approach [1], [2], [6]. A movement trajectory is not pre-defined or pre-planned, but generated during the movement from a differential equation. The dynamical system represents a whole flow field instead of a single-trajectory, like splines do. Thus, a suitably chosen flow field can automatically correct for perturbations of the state and guarantee convergence to a goal state. Different from the above mentioned continuous control, only a few control commands need to be specified, and the differential equation generates the corresponding trajectory, reducing the control load.

A realization of such dynamical systems are *dynamic movement primitives* [1], [2], [6]. In this framework, differential equations are adapted to generate given movements. Together these equations can form a library of movement primitives. Despite these possibilities, a weak point of this framework has been the generalization to new movement targets; this generalization could be unnatural and pose technical difficulties for the robotic execution (Section II).

In the present manuscript, we introduce a new formulation that overcomes these problems. Our new equations are motivated from neurophysiology. In frog, when stimulating the spinal cord and measuring the resulting forces at the foot for different leg postures, a force field can be constructed [7]. These fields were suggested as building blocks for movement generation [8]. Based on properties of these force fields, we derive our new dynamic equations. These equations generalize movements to new targets in a way that overcomes problems of the previous formulation and results in a more human-like movement adaptation.

Furthermore, different from previous work [1], [2], we use dynamical systems to generate trajectories in the task space of a robot, while the robot's joints are controlled using inverse kinematics and dynamics [9]. This approach has several advantages: manipulated objects and obstacles are in the same space as the dynamical system; we have no correspondence problem between a human demonstrator and the robot, and the movement generation is not restricted to a specific robot.

Finally, we extend our dynamical systems to include obstacle avoidance. In a previous study, we added to our dynamic equations the gradient of a potential field around an obstacle [10]. Potential fields are often used for obstacle avoidance [11], but may lead to local minima if several obstacles are present [12]. Here, we add a term that empirically describes how humans steer around obstacles [13]. For this term, we prove that our dynamical system converges at the desired goal position even for arbitrarily-many obstacles.

The remainder of this article is organized as follows. Section II describes the original form of dynamic movement primitives and mentions advantages and shortcomings. Section III presents the new formulation, which overcomes these shortcomings, and describes the motivation from biology. Section IV shows the extension of our dynamical systems for obstacle avoidance. A proof is presented that

This research was supported in part by DFG grant HO 3887/1-1, National Science Foundation grants ECS-0325383, IIS-0312802, IIS-0082995, ECS-0326095, ANI-0224419, the DARPA program on Learning Locomotion, and the ATR Computational Neuroscience Laboratories.

All authors were with the Department of Computer Science, University of Southern California, USA. Currently, H.H. is with the Department of Biomedical Engineering at USC. heiko@clmc.usc.edu

demonstrates convergence of the movement to the desired goal in the presence of obstacles. Section V demonstrates our framework with the Sarcos Slave robot arm. We show real-time adaptation to changing movement targets and to moving obstacles.

II. DYNAMIC MOVEMENT PRIMITIVES

We first describe dynamic movement primitives (DMPs) for a one-dimensional state variable and, then, show how many-dimensional systems are dealt with.

A. One-dimensional system

A DMP generates a movement trajectory $x(t)$ with velocity $v(t)$. The equations of motion are motivated from the dynamics of a damped spring attached to a goal position g and perturbed by a non-linear acceleration:

$$\tau \dot{v} = K(g-x) - Dv + (g-x_0)f(s) \quad (1)$$

$$\tau \dot{x} = v \ , \quad (2)$$

where x_0 is the start point of a movement, K the spring constant, D a damping constant, τ a constant scaling factor of the movement duration, and f a parametrized non-linear function. The function f is defined as

$$f(s) = \frac{\sum_i \psi_i(s) w_i}{\sum_i \psi_i(s)} s \ , \quad (3)$$

with Gaussian functions $\psi_i(s) = \exp(-h_i(s-c_i)^2)$. Instead of time, f depends explicitly on a phase variable s,

$$\tau \dot{s} = -\alpha \, s \ , \quad (4)$$

where α is a predefined constant. The advantage of using (4) instead of making f directly depend on time is that we can control the movement duration just by changing τ without changing the movement trajectory: the differential equations are scale-invariant about time. The state variable s is initially set to 1. In the limit $s \to 0$, the function f vanishes and (1) becomes a linear spring equation that converges to g. Equation (4) has been called 'canonical system'.

To reproduce a desired trajectory, we adapt the parameters w_i. For this adaptation, the following four steps are carried out: first, a movement $x(t)$ is recorded and its derivatives $v(t)$ and $\dot{v}(t)$ are computed for each time step t. Second, $f(t)$ is computed based on (1). Third, (4) is integrated and $s(t)$ evaluated. Forth, using these time arrays, we find the weights w_i in (3) by linear regression, which can be solved efficiently.

Dynamic movement primitives have several favorable features:

+ Any smooth movement can be generated with a DMP.
+ The differential equations converge to the goal point g and automatically adapt to perturbations of the state x. This flexibility is an advantage over a functional representation of a movement trajectory, like in splines.
+ The generated movement adapts online to a change of the goal g.
+ The differential equations are translation invariant.
+ By setting τ, we can choose the duration of a movement without changing the movement trajectory.

However, this formalism has also shortcomings:

- If start and end-point of a movement are the same, $x_0 = g$, no movement will be generated, $\dot{v}(t) = 0$.
- If g is close to x_0, a small change in g may lead to huge accelerations (Fig. 2) that break the limits of the robot.
- If changing g across the zero point, the whole movement inverts (Fig. 2).

In this article, we introduce a new formulation of dynamic equations, which overcomes these shortcoming and keeps the favorable features mentioned above. A further problem that we address arises for many-dimensional systems.

B. Many-dimensional system

The above equations are written for a one-dimensional system. For many dimensions, each dimension can be described by equation (1) and each dimension has its own set of parameters w_i. For all dimensions, the same phase variable s can be used. Thus, a single canonical system drives the non-linear functions for all dimensions.

We want to apply DMPs to describe the end-effector motion in three-dimensional task space. For this application, we need to formulate the equation of motion (1) for each dimension. However, the behavior of the differential equations will depend on the orientation of the coordinate system in task space (Fig. 1). Such a dependency is undesired. The new DMP version that we are going to present in this article is invariant under the choice of coordinate system.

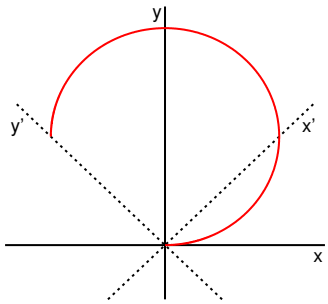

Fig. 1. Original form of dynamic movement primitives depends on choice of coordinate system. Two coordinate systems are plotted (x,y) and (x',y'). In the primed coordinate system, start and end point of a sample movement (red curve) are on the y'-axis. In this case, the motion generation will fail to produce the desired trajectory, since the motion will be restricted to $x' = 0$.

III. NEW FORM OF DMP

We motivate our new dynamical-systems equations from neurophysiology, show invariance under invertible affine transformations, and demonstrate the generalization to new movement targets.

A. Motivation from neurophysiology

To motivate our new equations for movement generation, we use three key findings of spinal force fields in frog [7]:

1) These force fields are often convergent.

2) The magnitude of force fields is modulated in time by bell-shaped time pulses.
3) Simultaneously stimulated force fields add up linearly.

These findings suggest that a limb moves along a trajectory, by activating different convergent force fields in sequence [7]. Here, we build a differential equation based on a sequence of convergent fields. However, instead of force, we use acceleration fields. Thus, as in the original form of DMP, we use a separate inverse-dynamics controller to obtain the control torques.

We describe the acceleration fields in 3D task space. These fields are a function of the end-effector position $\mathbf{x}$ and its velocity $\mathbf{v}$. For simplicity, we use linear fields

$$\mathbf{a}_i(\mathbf{x}, \mathbf{v}) = \mathbf{K}(\mathbf{w}_i - \mathbf{x}) - \mathbf{D}\mathbf{v} \quad , \tag{5}$$

which are centered at $\mathbf{w}_i$. Here, we assume that no external forces act on the limb, apart from constant forces, like gravity. Equation (5) can be interpreted as a linear approximation of an arbitrary differentiable acceleration field (Taylor expansion up to first term). Moreover, (5) describes the dynamics of a mass point (mass=1) that is connected with a linear spring to a point at $\mathbf{w}_i$, having spring constant $\mathbf{K}$ and damping constant $\mathbf{D}$.

Each field is modulated over time with a Gaussian function centered at time c_i,

$$\psi_i(t) = \exp\left(-h_i(t - c_i)^2\right) \quad . \tag{6}$$

Using this time modulation and the above mentioned summation property, we obtain a new more complex field,

$$\mathbf{a}(\mathbf{x}, \mathbf{v}, t) = \frac{\sum_i \psi_i(t)\mathbf{a}_i(\mathbf{x}, \mathbf{v})}{\sum_i \psi_i(t)} \quad . \tag{7}$$

Substituting (5) in (7) and setting $\dot{\mathbf{v}} = \mathbf{a}$ results in the equations of motion

$$\dot{\mathbf{v}} = \mathbf{K}\left(\frac{\sum_i \psi_i(t)\mathbf{w}_i}{\sum_i \psi_i(t)} - \mathbf{x}\right) - \mathbf{D}\mathbf{v} \tag{8}$$

$$\dot{\mathbf{x}} = \mathbf{v} \ . \tag{9}$$

To make the equation of motion converge to the goal $\mathbf{g}$, we add around $\mathbf{g}$ another linear convergent field (5) and shift the weight from (8) to the new field. As weight, we use the phase variable s, as computed by (4), and thus, the new acceleration field becomes

$$\begin{aligned} \dot{\mathbf{v}} &= s\,\mathbf{K}\left(\frac{\sum_i \psi_i(s)\mathbf{w}_i}{\sum_i \psi_i(s)} + \mathbf{x}_0 - \mathbf{x}\right) \\ &+ (1-s)\mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{D}\mathbf{v} \ . \end{aligned} \tag{10}$$

We inserted an extra $\mathbf{x}_0$ to make the equation translation invariant. Furthermore, we changed the dependence of ψ on t to s, as in the original DMP. Thus, we use the same canonical system (4). Equation (10) can be rewritten into

$$\dot{\mathbf{v}} = \mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{D}\mathbf{v} - \mathbf{K}(\mathbf{g} - \mathbf{x}_0)s + \mathbf{K}\mathbf{f}(s) \quad . \tag{11}$$

This equation is similar to the original form of DMP - see (1). The function $\mathbf{f}(s)$ is the same as defined in (3). In the limit $s \to 0$, both forms are the same and converge to $\mathbf{g}$.

Apart from (11), the formalism of our modified DMP is the same as the original DMP; the constant τ can be multiplied to the left side of (11). As before, the parameters $\mathbf{w}_i$ are learned by computing $\mathbf{f}(s)$ for a given desired trajectory.

B. Invariance under affine transformation

Different from the original DMP, the new form is invariant under rotations of the coordinate system. Generally, equation (11) is invariant under invertible affine transformations of $\mathbf{x}$ and $\mathbf{v}$. To demonstrate this invariance, we substitute

$$\begin{aligned} \mathbf{x} &= \mathbf{S}\mathbf{x}' & \mathbf{x}_0 &= \mathbf{S}\mathbf{x}'_0 \\ \mathbf{v} &= \mathbf{S}\mathbf{v}' & \mathbf{K} &= \mathbf{S}\mathbf{K}'\mathbf{S}^{-1} \\ \dot{\mathbf{v}} &= \mathbf{S}\dot{\mathbf{v}}' & \mathbf{D} &= \mathbf{S}\mathbf{D}'\mathbf{S}^{-1} \\ \mathbf{g} &= \mathbf{S}\mathbf{g}' \ , \end{aligned} \tag{12}$$

where $\mathbf{S}$ is the invertible transformation matrix. To adapt the parameters $\mathbf{w}_i$ for generating a desired trajectory $\mathbf{x}(t)$, we need to compute $\mathbf{f}(s)$. After substituting (12) into (11), the non-linear function $\mathbf{f}$ becomes

$$\begin{aligned} \mathbf{f} &= \mathbf{S}\left(\mathbf{K}'^{-1}\dot{\mathbf{v}}' - (\mathbf{g}' - \mathbf{x}') + \mathbf{K}'^{-1}\mathbf{D}'\mathbf{v}' + (\mathbf{g}' - \mathbf{x}'_0)s\right) \\ &:= \mathbf{S}\mathbf{f}' \ . \end{aligned} \tag{13}$$

Substituting (12) and $\mathbf{f} = \mathbf{S}\mathbf{f}'$ into (11), we obtain for the prim variables the same equation as (11). Thus, we have shown invariance under the above transformation.

C. Movement generalization to new target

To adapt a movement to a new goal $\mathbf{g}$, we just change $\mathbf{g}$ in (11), and the whole trajectory adapts accordingly. This parameter may be changed at the beginning of a movement or even during the movement. Compared to the original form, movement adaptation improved (Fig. 2). Critically, the non-linear term does not scale anymore with $(\mathbf{g} - \mathbf{x}_0)$, avoiding some of the previous shortcomings (see Section II).

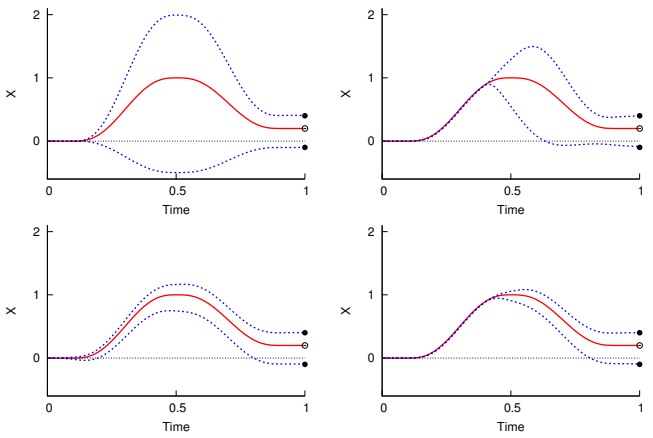

Fig. 2. Comparison of goal adaption between original (top row) and new (bottom) form of dynamic movement primitives. Graphs show a sample movement $x(t)$ to the original goal (red solid curve) and its adaptation to new goals (blue dashed curves). As the time of goal switch, we chose either 0 (left column) or 0.4 (right column).

Furthermore, we tested how the adaptation to a changing goal $\mathbf{g}$ in the new equation compares with human behavior.

In a separate experiment [14], some of us recorded human movements with a stylus on a graphics tablet. Subjects made curved movements towards a target displayed on a screen. During half of the trials (200ms after movement onset), the target jumped to a different location, and we observed how the subjects adapted their movement to the new goal. The observed movement adaptation could be explained by changing $\mathbf{g}$ in equation (11) - see Fig. 3 and [14]. Here, we adapted the parameters $\mathbf{w}_i$ such that the differential equation reproduced a subject's average movement to the original goal.

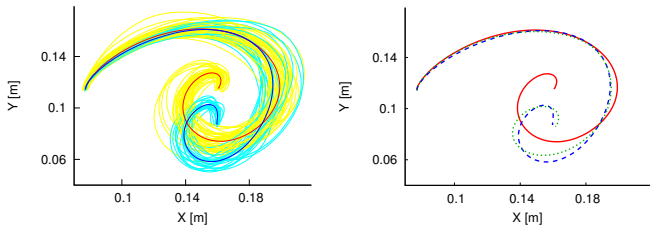

Fig. 3. Goal adaptation of the new DMP equations compared to human behavior. (Left) Trajectories on a graphics tablet for one subject: raw trajectories (yellow and cyan) and their means (red and blue) are shown. The red curve is the movement to the original goal, and the blue curve is the adaptation to the switching target. (Right) The adaptation of our dynamical system to the new target (green dotted) is compared with the experimental data (blue dashed).

IV. OBSTACLE AVOIDANCE

We exploit the robustness of dynamical systems against perturbations for obstacle avoidance. To account for the avoidance behavior, an additional term $\mathbf{p}(\mathbf{x}, \mathbf{v})$ is added to our differential equation (11),

$$\dot{\mathbf{v}} = \mathbf{K}(\mathbf{g}-\mathbf{x}) - \mathbf{D}\mathbf{v} - \mathbf{K}(\mathbf{g}-\mathbf{x}_0)s + \mathbf{K}\mathbf{f}(s) + \mathbf{p}(\mathbf{x}, \mathbf{v}) \,. \quad (14)$$

We first consider one obstacle with fixed position, then, many obstacles, and, finally, moving obstacles.

A. Single static obstacle

Fajen and Warren [13] found a differential equation that models human obstacle avoidance. Their equation describes the steering angle φ (Fig. 4), which is modeled to change according to

$$\dot{\varphi} = \gamma\, \varphi \exp(-\beta\, |\varphi|) \quad . \quad (15)$$

For illustration, (15) is plotted in Fig. 5. For large angles, $\dot{\varphi}$ approaches zeros, i.e., a movement away from the obstacle needs no correction. We combine equation (15) with

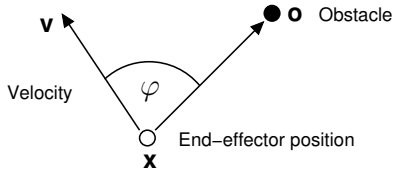

Fig. 4. Definition of the steering angle φ.

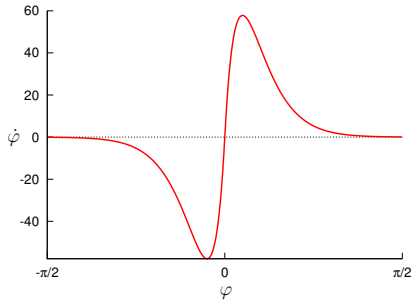

Fig. 5. Change of steering angle φ. Here, the parameters $\gamma = 1000$ and $\beta = 20/\pi$ are used, the same as in the robot experiments.

our dynamic movement primitives. The change in steering direction changes the velocity vector $\mathbf{v}$ as follows,

$$\dot{\mathbf{v}} = \mathbf{R}\mathbf{v}\dot{\varphi} \quad , \quad (16)$$

where $\mathbf{R}$ is a rotational matrix with axis $\mathbf{r} = (\mathbf{o} - \mathbf{x}) \times \mathbf{v}$ and angle of rotation of $\pi/2$; vector $\mathbf{o}$ is the position of the obstacle. Equation (16) can be derived by writing $\mathbf{v}$ as $\mathbf{v} = [v\cos(\varphi); v\sin(\varphi)]$ in the plane spanned by $(\mathbf{o}-\mathbf{x})$ and $\mathbf{v}$ and by deriving this expression with respect to time.

We append the obstacle induced change in velocity as extra term to our dynamic motion equation; thus we choose

$$\mathbf{p}(\mathbf{x}, \mathbf{v}) = \gamma \mathbf{R}\mathbf{v}\varphi \exp(-\beta\varphi) \quad , \quad (17)$$

with $\varphi = \cos^{-1}((\mathbf{o} - \mathbf{x})^T\mathbf{v}/(|\mathbf{o} - \mathbf{x}| \cdot |\mathbf{v}|))$; this value is always positive. Obstacle-avoidance movements with this extended DMP are shown in Fig. 6.

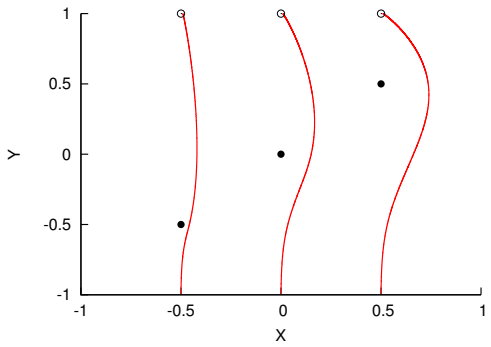

Fig. 6. Obstacle avoidance in 2D space. Three separate movements are shown, each with a different obstacle position (black dot). Goal positions are marked by circles. Here, for simplicity, $\mathbf{f}(s) = (\mathbf{g} - \mathbf{x}_0)s$; thus, terms depending on s vanish. The goal positions were shifted by 0.01 to the right from the lines going through start and obstacle positions.

In the following, we show that (14) with (17) converges to the goal position $\mathbf{g}$. First, we demonstrate convergence for one obstacle, and, then, extend to many obstacles. For $t \to \infty$, the terms in (14) that depend on s approach 0 exponentially; thus, we just need to study convergence of the reduced equation

$$\dot{\mathbf{v}} = \mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{D}\mathbf{v} + \gamma\mathbf{R}\mathbf{v}\varphi\exp(-\beta\varphi) \quad . \quad (18)$$

The state $[\mathbf{x}; \mathbf{v}] = [\mathbf{g}; 0]$ is a stationary point in the equation. All other states converge to this point, which we will show by constructing a Lyapunov function [15]. As Lyapunov

function candidate $V(\mathbf{x}, \mathbf{v})$, we use the energy of the linear spring system, $\dot{\mathbf{v}} = \mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{D}\mathbf{v}$ (here, using unit mass),

$$V(\mathbf{x}, \mathbf{v}) = \frac{1}{2}(\mathbf{g} - \mathbf{x})^T \mathbf{K}(\mathbf{g} - \mathbf{x}) + \frac{1}{2}\mathbf{v}^T\mathbf{v} \ . \qquad (19)$$

To prove convergence, we need to show $\dot{V} < 0$ for $\mathbf{v} \neq 0$,

$$\begin{aligned} \dot{V} &= \nabla_{\mathbf{x}} V^T \dot{\mathbf{x}} + \nabla_{\mathbf{v}} V^T \dot{\mathbf{v}} \\ &= -(\mathbf{g} - \mathbf{x})^T \mathbf{K}\mathbf{v} + \mathbf{v}^T \dot{\mathbf{v}} \\ &= -\mathbf{v}^T \mathbf{K}(\mathbf{g} - \mathbf{x}) + \mathbf{v}^T \mathbf{K}(\mathbf{g} - \mathbf{x}) - \mathbf{v}^T \mathbf{D}\mathbf{v} \\ &\quad + \gamma \mathbf{v}^T \mathbf{R}\mathbf{v} \varphi \exp(-\beta\varphi) \\ &= -\mathbf{v}^T \mathbf{D}\mathbf{v} < 0 \ . \end{aligned} \qquad (20)$$

The damping matrix $\mathbf{D}$ is negative definite by choice to guarantee damping. The term $\mathbf{v}^T\mathbf{R}\mathbf{v}$ is 0 since the matrix $\mathbf{R}$ is a rotation by 90 degrees. If $\mathbf{v} = 0$ and $\mathbf{x} \neq \mathbf{g}$, then $\dot{V} = 0$; however, if $\mathbf{x} \neq \mathbf{g}$ then $\dot{\mathbf{v}} \neq 0$, and $\dot{V}$ changes. Thus, according to LaSalle's theorem [15], $\mathbf{x}$ converges to $\mathbf{g}$. Therefore, we have shown that our extended DMP equations converge globally to the goal position $\mathbf{g}$.

B. Many static obstacles

This convergence is further guaranteed if we have more than one obstacle. Here, the perturbation term $\mathbf{p}(\mathbf{x}, \mathbf{v})$ becomes

$$\mathbf{p}(\mathbf{x}, \mathbf{v}) = \gamma \sum_i \mathbf{R}_i \mathbf{v} \varphi_i \exp(-\beta\varphi_i) \,, \qquad (21)$$

where $\varphi_i = \cos^{-1}((\mathbf{o}_i - \mathbf{x})^T\mathbf{v}/(|\mathbf{o}_i - \mathbf{x}| \cdot |\mathbf{v}|))$. Since each term $\mathbf{v}^T\mathbf{R}_i\mathbf{v}$ vanishes, the above energy term (19) is again a Lyapunov function of the system. Figure 7 shows sample solutions for many obstacles.

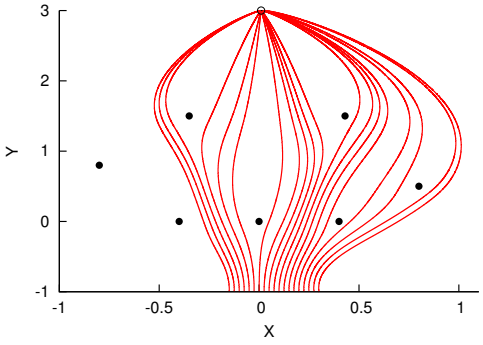

Fig. 7. Avoidance of multiple obstacles (black dots). Trajectories for different start positions are shown. Here, for simplicity, $\mathbf{f}(s) = (\mathbf{g} - \mathbf{x}_0)s$.

C. Moving obstacles

We can extend our equations to moving obstacles. To avoid collisions, we need to consider the relative velocity between the end-effector and the obstacle, not just the end-effector velocity itself. Therefore, for each obstacle i, we compute φ_i in the reference frame of the obstacle,

$$\varphi_i = \cos^{-1}\left(\frac{(\mathbf{o}_i - \mathbf{x})^T(\mathbf{v} - \dot{\mathbf{o}}_i)}{|\mathbf{o}_i - \mathbf{x}| \cdot |\mathbf{v} - \dot{\mathbf{o}}_i|}\right) \ , \qquad (22)$$

where $\dot{\mathbf{o}}_i$ is the velocity of obstacle i. Equation (16) needs to be adapted as well to the relative velocity; thus, the adapted $\mathbf{p}(\mathbf{x}, \mathbf{v})$ becomes

$$\mathbf{p}(\mathbf{x}, \mathbf{v}) = \gamma \sum_i \mathbf{R}_i(\mathbf{v} - \dot{\mathbf{o}}_i)\varphi_i \exp(-\beta\varphi_i) \,. \qquad (23)$$

The above convergence proof does not hold for this $\mathbf{p}(\mathbf{x}, \mathbf{v})$ because $\mathbf{v}^T\mathbf{R}_i\dot{\mathbf{o}}_i$ does not vanish. However, in most cases, for $t \to \infty$, the obstacles will either move away from the robot ($\mathbf{p}(\mathbf{x}, \mathbf{v}) = 0$) or come to a standstill (reduction to static case). Thus, the robotic motion will converge again.

V. ROBOT DEMONSTRATION

In a robot experiment, we show the utility of our new framework by demonstrating movement adaptation to changing targets and obstacle avoidance.

A. Methods

We used a ten degree-of-freedom (DOF) Sarcos anthropomorphic robot arm (Fig. 8 and 9). This robot has seven DOF for the arm and three for the fingers.

With our dynamical system, we describe the motion of the hand, the end-effector. To compute the control torques at the arm joints, we used inverse kinematics and dynamics [9]; the inverse kinematics took also into account the orientation of the hand [10]. The control loop ran at 480 Hz.

We obtained the desired end-effector motion from human demonstration. A demonstrator moved an exoskeleton robot arm (Sarcos Master arm). In this article, we demonstrated the motion of placing a cup on a table. The recorded motion was used to compute the parameters $\mathbf{w}_i$ of the non-linear function $\mathbf{f}(s)$ in the dynamical-systems equations.

For the parameters in our dynamical system, we made the following choices. The centers c_i of $\psi_i(s)$ were fixed and logarithmically distributed between 0.001 and 1. We set the bandwidth parameters h_i to $h_i = 0.5(c_i - c_{i-1})^{-2}$. The decay factor α was chosen such that $s = 0.001$ at the end of a movement. As spring constant $\mathbf{K}$, we used a diagonal matrix with $K_{ii} = 150$. The damping D was chosen to make the system critically damped, $D_{ii} = 2\sqrt{K_{ii}}$. For obstacle avoidance, we used the parameters $\beta = 20/\pi$ and $\gamma = 1000$.

To test online adaption in our robot setup, we used a stereo-vision system to obtain the position of the obstacle and the target (60 Hz sampling rate). The vision system extracted color blobs: we used a blue ball as obstacle and a green coaster as target (final position of the placing movement). The location of these color blobs was transformed into Cartesian-space coordinates. The obtained positions of target, $\mathbf{g}$, and obstacle, $\mathbf{o}$, were updated in real time in our differential equations.

B. Results

The robot could reproduce a demonstrated movement and generalize it to a new target by changing only $\mathbf{g}$ in the differential equations (Fig. 8). If the target changed during the movement, the end-effector motion adapted to this new target (see video supplement). In addition, the robot avoided a static obstacle in its path, which was hit when our obstacle avoidance was switched off (video supplement). For

a moving obstacle, the robot corrected its end-effector motion away from the approaching obstacle, and, still, this motion converged at the desired goal (Fig. 9, video supplement).

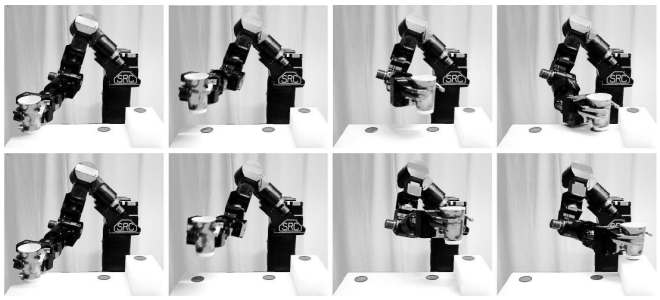

Fig. 8. Placing a cup with the Sarcos Slave arm. The first row shows the reproduction of a demonstrated movement. The second row shows the generalization to a new target position.

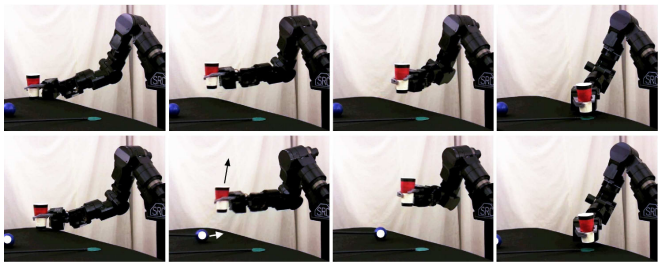

Fig. 9. Avoiding an approaching obstacle (white disc) - see movie supplement. The first row shows the motion without obstacle and the second row with moving obstacle.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

To generate robotic movements, dynamic movement primitives are an alternative to preplanned and precomputed trajectories. Movements are generated from differential equations, and the attractor dynamics of these equations allow the automatic adaptation of a movement trajectory to changing targets and obstacles in the way.

In this article, we presented new equations that overcome problems of the original DMP formulation. Our modified DMP is invariant with respect to rotations of the coordinate system in task space and generalizes movements to new targets more like humans do. The new equations were motivated from biology: they were derived from properties of convergent force fields in frogs.

Furthermore, we added obstacle avoidance by adding an acceleration term to the equation of motion. This term was taken from the literature and describes empirically how humans steer around obstacles. Together with the attractor dynamics, this term allowed our robot without re-planning to avoid obstacles, while converging to the original target. We proved this convergence for arbitrarily-many obstacles.

The feasibility of our new framework was demonstrated using a Sarcos robot arm. First, we recorded a desired motion by moving a robotic exoskeleton; alternatively, a visual or magnetic motion tracking system might be used. Second, we adapted our dynamical system to generate the demonstrated motion. Third, the robot tracked the solution of our equation of motion, while it was integrated. Since we decoupled trajectory generation and robot control, our robot could be easily replaced by any robotic manipulator with at least six DOF and sufficient working range. Finally, the possibility to cover a range of motions by changing only a few parameters, like the goal position, will probably simplify the control of full-body humanoid robots and prosthetic arms.

B. Future Works

In future work, we will apply our framework to a full-body humanoid robot (Sarcos CB). Furthermore, we aim to control prosthetic arms. Here, the human wearer will first choose a movement primitive from a library, focus on the movement goal, while an eye-tracker extracts the fixation point, and then, our dynamical system will generate the whole movement, including automatic obstacle avoidance.

REFERENCES

[1] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *IEEE International Conference on Robotics and Automation*, Washington, DC, 2002, pp. 1398–1403.

[2] ——, "Learning attractor landscapes for learning motor primitives." in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds., vol. 15. MIT Press, Cambridge, MA, 2003, pp. 1523–1530.

[3] G. Cheng, S.-H. Hyon, J. Morimoto, A. Ude, G. Colvin, W. Scroggin, and S. Jacobsen, "CB: A humanoid research platform for exploring neuroscience," in *IEEE International Conference on Humanoid Robots*, Genova, 2006, pp. 182–187.

[4] S. Adee, "Dean Kamen's 'Luke Arm' prosthesis readies for clinical trials," 2008. [Online]. Available: http://www.spectrum.ieee.org/feb08/5957

[5] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, pp. 1098–1101, 2008.

[6] S. Schaal, *Dynamic Movement Primitives -A Framework for Motor Control in Humans and Humanoid Robotics*. Springer Tokyo, 2005, ch. 6, pp. 261–280.

[7] S. F. Giszter, F. A. Mussa-Ivaldi, and E. Bizzi, "Convergent force fields organized in the frog's spinal cord," *Journal of Neuroscience*, vol. 13, no. 2, pp. 467–491, 1993.

[8] E. Bizzi, F. A. Mussa-Ivaldi, and S. F. Giszter, "Computations underlying the execution of movement: A biological perspective," *Science*, vol. 253, no. 5017, pp. 287–291, 1991.

[9] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: a theoretical and emprical comparison," *International Journal of Robotics Research*, vol. 27, pp. 737–757, 2008.

[10] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *IEEE International Conference on Humanoid Robots*, Daejeon, Korea, 2008.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and fast mobile robots," *International Journal of Robotics Research*, vol. 5, p. 90, 1986.

[12] F. Janabi-Sharifi and D. Vinke, "Integration of the artificial potential field approach with simulated annealing for robot path planning," *Proceedings of the IEEE International Symposium on Intelligent Control*, pp. 536–541, Aug 1993.

[13] B. R. Fajen and W. H. Warren, "Behavioral dynamics of steering, obstacle avoidance, and route selection," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 29, no. 2, pp. 343–362, 2003.

[14] H. Hoffmann and S. Schaal, "Human movement generation based on convergent flow fields: a computational model and a behavioral experiment," in *Advances in Computational Motor Control VI*, R. Shadmehr and E. Todorov, Eds., San Diego, CA, 2007.

[15] H. K. Khalil, *Nonlinear Systems*. Prentice Hall, 2002.