Robotics and Autonomous Systems 62 (2014) 752–765

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions

S. Mohammad Khansari-Zadeh *, Aude Billard [1]

EPFL-STI-IMT-LASA, CH-1015, Lausanne, Switzerland

HIGHLIGHTS

- Proposing a new parameterization to model complex Lyapunov functions.
- Estimating task-oriented Lyapunov functions from demonstrations.
- Ensuring stability of nonlinear autonomous dynamical systems.
- Applicability to any smooth regression method.

ARTICLE INFO

Article history:
Received 2 September 2013
Received in revised form
20 February 2014
Accepted 3 March 2014
Available online 13 March 2014

Keywords:
Robot point-to-point movements
Imitation learning
Control Lyapunov function
Nonlinear dynamical systems
Stability analysis
Movement primitives

ABSTRACT

We consider an imitation learning approach to model robot point-to-point (also known as discrete or reaching) movements with a set of autonomous Dynamical Systems (DS). Each DS model codes a behavior (such as reaching for a cup and swinging a golf club) at the kinematic level. An estimate of these DS models are usually obtained from a set of demonstrations of the task. When modeling robot discrete motions with DS, ensuring stability of the learned DS is a key requirement to provide a useful policy. In this paper we propose an imitation learning approach that exploits the power of Control Lyapunov Function (CLF) control scheme to ensure global asymptotic stability of nonlinear DS. Given a set of demonstrations of a task, our approach proceeds in three steps: (1) Learning a valid Lyapunov function from the demonstrations by solving a constrained optimization problem, (2) Using one of the-state-of-the-art regression techniques to model an (unstable) estimate of the motion from the demonstrations, and (3) Using (1) to ensure stability of (2) during the task execution via solving a constrained convex optimization problem. The proposed approach allows learning a larger set of robot motions compared to existing methods that are based on quadratic Lyapunov functions. Additionally, by using the CLF formalism, the problem of ensuring stability of DS motions becomes independent from the choice of regression method. Hence it allows the user to adopt the most appropriate technique based on the requirements of the task at hand without compromising stability. We evaluate our approach both in simulation and on the 7 degrees of freedom Barrett WAM arm.

1. Introduction

When designing robots meant to interact in a human environment, it is essential to develop methods that can easily transfer skills from nonexpert users to robots, and at the same time, provide the required robustness and reactivity to interact with a dynamic environment. Classical approaches to modeling robot motions rely on decomposing the task execution into two separate processes: *planning* and *execution* [1]. The former is used as a means to generate a feasible path that can satisfy the task's requirements, and the latter is designed so that it follows the generated feasible path as closely as possible. Hence these approaches consider any deviation from the desired path (due to perturbations or changes in environment) as the tracking error, and various control theories have been developed to efficiently suppress this error in terms of some objective functions. Despite the great success of these approaches in providing powerful robotic systems, particularly in factories, they are ill-suited for robotic systems that are aimed to work in the close vicinity of humans, and thus alternative techniques must be sought.

* Correspondence to: Ecole Polytechnique Federale de Lausanne, LASA Laboratory, Switzerland. Tel.: +41 21 693 5462.
E-mail addresses: mohammad.khansari@epfl.ch (S. Mohammad Khansari-Zadeh), aude.billard@epfl.ch (A. Billard).
URL: http://lasa.epfl.ch/.
1 Ecole Polytechnique Federale de Lausanne, LASA Laboratory, Switzerland.

http://dx.doi.org/10.1016/j.robot.2014.03.001
0921-8890/

In robotics, Dynamical Systems (DS) based approaches to motion generation have been shown to be interesting alternatives to classical methods as they offer a natural means to integrate planning and execution into one single unit [2–5]. For instance when modeling robot reaching motions with DS, all possible solutions to reach the target are embedded into one single model [6]. Such a model represents a global map which specifies *instantly* the correct direction for reaching the target, considering the current state of the robot, the target, and all the other objects in the robot's working space. Such models are more similar to human movements in that they can effortlessly adapt its motion to changes in the environment rather than stubbornly following the previous path [6–12]. In other words, the main advantage of using DS-based formulation can be summarized as: "Modeling movements with DS allows having robotic systems that have inherent adaptivity to changes in a dynamic environment, and that can swiftly adopt a new path to reach the target". This advantage is the direct outcome of having a unified planning and execution unit.

Imitation learning (also known as learning from demonstrations) is one of the most common and intuitive ways of building an estimate of DS motions from a set of demonstrations [6,13,7,8]. When modeling robot motions with DS, *ensuring stability* of the estimated DS – ensuring the robot reaches the final desired state – is one of the main challenges that should be addressed in order to obtain a useful control policy. This is by construction a difficult conundrum since one needs to deal with the problem of both estimating a nonlinear function, and additionally ensuring stability of an unknown nonlinear DS. Most imitation learning approaches tackle the stability problem by using a time (phase)-dependent clock that smoothly switches from an unstable nonlinear DS to a globally asymptotically stable linear system [7,8,12]. However due to the time-dependency, these approaches are sensitive to perturbations. In our previous work [6], we have presented a formal stability analysis to ensure global asymptotic stability of autonomous (i.e. time-invariant) DS. In order to derive stability conditions, that work relied on two assumptions: (1) Robot motions are formulated with Gaussian Mixture Regression (GMR) [14], and more importantly (2) The energy of the motion takes a quadratic form. The former constraints the user to using solely a specific regression method (namely GMR), while the latter limits the range of possible motions that can be modeled accurately.

There are numerous nonlinear regression techniques to estimate nonlinear DS. Each of these techniques has its own pros and cons which make their use very task-dependent. For instance, Gaussian Process Regression (GPR) [15] is an accurate method but is computationally expensive. GMR is computationally fast, but is comparatively less accurate, for instance, than GPR. Locally Weighted Projection Regression (LWPR) [16] is a powerful tool for incremental learning but requires setting several initial parameters. Various advantages and disadvantages can also be observed when using other techniques such as Support Vector Regression (SVR) [17], Reservoir Computing (RC) [18], and Gaussian Process Latent Variable Model (GPLVM) [19]. Thus, it would be advantageous if one could freely choose the most appropriate regression technique based on requirements of the task at hand, while still ensuring the robot can reach the desired target point. Note that the standard training of the above regression techniques do not ensure global asymptotic stability at the target [6].

The field of control theory has provided us with various tools to design a stable controller for (nonlinear) DS around a desired target point. Control Lyapunov Function (CLF) control scheme [20–22] is one of these techniques that is designed based on the following intuitive idea: "Associate a Lyapunov function (i.e. energy function) $V(\boldsymbol{\xi})$ with its global minimum at the target point $\boldsymbol{\xi}^*$ to the DS $\boldsymbol{f}(\boldsymbol{\xi})$ that needs to be stabilized. At each time step, apply a stabilizing (or control) command $\boldsymbol{u}(\boldsymbol{\xi})$ so as to force $V(\boldsymbol{\xi})$ decreases. This system is guaranteed to asymptotically reach the target point".

In this paper we propose an imitation learning approach that exploits the power of Control Lyapunov Function (CLF) control scheme to ensure global asymptotic stability of DS-based robot reaching motions. For brevity, we call the proposed approach CLF-DM (Control Lyapunov Function-based Dynamic Movements). Given a set of demonstrations of a task, our approach proceeds in three steps: (1) Learning a valid Lyapunov function from a set of demonstrations, (2) Using one of the-state-of-the-art regression techniques to model an (unstable) estimate of the motion from the demonstrations, and (3) Using (1) to ensure stability of (2) at the unique target point during the task execution. The proposed approach allows learning a larger set of robot motions compared to existing methods that are based on quadratic energy functions. Additionally, due to the CLF formalism, the problem of ensuring stability of DS motions becomes independent from the choice of regression method. Thus, one could now adopt the most appropriate technique based on the requirements of the task at hand without compromising stability. Furthermore, with the new approach, one has the possibility to have online/incremental learning which is crucial in many tasks as it allows the user to refine the robot motion in an interactive manner.

In contrast to classical CLF-based approaches where the energy function is mostly hand-tuned by the user, a non-trivial problem that has only been solved for special classes of DS [23], in our approach we build an estimate of the CLF from demonstrations. Additionally, by learning CLF, our approach is tailored to generate stabilizing commands that modify the unstable DS given by $\boldsymbol{f}(\boldsymbol{\xi})$ as least as possible at each iteration. Hence, it tries to maximize the similarity between the stabilized and the unstable DS which is crucial to generate motions that resemble the user demonstrations. Note that apart from the CLF approach, the Optimal Control techniques [24] can also be exploited so as to generate a sequence of stabilizing commands to ensure stability of $\boldsymbol{f}(\boldsymbol{\xi})$. Despite the successful realtime implementation of these approaches for linear DS, their implementation for nonlinear DS is still an open question and realtime solutions only exist for particular cases.

The contribution of this work are four-folds: (1) Proposing a new parameterization for energy function, called Weighted Sum of Asymmetric Quadratic Function (WSAQF), that significantly outperforms the classical way of modeling energy function with a quadratic function, (2) Presenting a constrained optimization problem to build an estimate of a task-oriented Lyapunov function from a set of demonstrations, (3) Proposing an optimal control problem (based on the learned Lyapunov function) to ensure stability of nonlinear autonomous DS, and (4) Extending the classical CLF control scheme and present it in the context of learning robot discrete motions from demonstrations. By taking this approach, we provide the user with the possibility to choose a regression technique of her choice (for example based on the requirements of the task at hand). We evaluate the proposed approach on a library of human handwriting motions and on the 7 degrees of freedom Barrett WAM arm.

2. Related work

The problem of motion generation for robot movement has been an active research topic in robotics for years, and many techniques have been suggested addressing different aspects of this problem. In this section we only focus on reviewing Dynamical System (DS) based approaches as it is the main topic covered in this work.[2] The DS approach to modeling robot motions is a type of

[2] Interested readers could refer to [25] for more information about the difference between DS-based approaches and other techniques such as path planners [26], time-indexed trajectory generators [27], and potential field methods [28].