7th. Int. Conf. on Soft Computing and Pattern Recognition (SocPar)

Incremental Learning of Reach-to-Grasp Behavior: A PSO-based Inverse Optimal Control Approach

Haitham El-Hussieny*†, Samy F. M. Assal *‡, A. A. Abouelsoud*§, Said M. Megahed*¶ and Tsukasa Ogasawara†
*Mechatronics and Robotics Engineering Department, School of Innovative Design Engineering
Egypt-Japan University of Science and Technology (E-JUST), Egypt
†Nara Institute of Science and Technology, 8916-5, Takayama, Ikoma, Nara 630-0192, Japan
‡on leave: Department of Production Engineering and Mechanical Design, Faculty of Engineering, Tanta University, Egypt
§on leave: Electronics and Communications Eng. Dept., Faculty of Engineering, Cairo University, Egypt
¶on leave: Mechanical Design and Production Engineering Department, Faculty of Engineering, Cairo University, Egypt
E-mail: {haitham.elhussieny, samy.assal, ahmed.ali, smegahed}@ejust.edu.eg, ogasawar@is.naist.jp

***Abstract*—In recent years, there has been an increasing interest in modeling natural human movements. The main question to be addressed is: what is the optimality criteria that human has optimized to achieve a certain movement. One of the most significant current discussions is the modeling of the reach-to-grasp movements that human naturally perform while approaching a certain object for grasping. Recent advances in Inverse Reinforcement Learning (IRL) approaches have facilitated investigation of reach-to-grasp movements in terms of the optimal control theory. IRL aims to learn the cost function that best describes the demonstrated human reach-to-grasp movements. Thus far, gradient-based techniques have been used to obtain the parameters of the underlying cost function. Such approaches, however, have failed to find the global optimal parameters since they are limited by locating only local optimum values. In this research, learning of the cost function for the reach-to-grasp movements is addressed as an Inverse Linear Quadratic Regulator (ILQR) problem, where linear dynamic equations and a quadratic cost are assumed. An efficient evolutionary optimization technique, Particle Swarm Optimization (PSO), is used to obtain the unknown cost for the reach-to-grasp movements under consideration. Moreover, an incremental-ILQR Algorithm is proposed to adjust the learned cost once new untrained demonstrations exist to overcome the over-fitting issue. The obtained results are encouraging and show harmony with those in neuroscience literature.**

Keywords; Inverse reinforcement learning, Linear Quadratic Regulator, Reach-to-grasp modeling, Particle Swarm Optimization, Incremental learning.

I. Introduction

Modeling the daily human motor movements in certain contexts has become the central issue for human centered applications. Particularly, for the reach-to-grasp movements, there is a growing body of literature that recognizes how the understanding of human reach-to-grasp behavior is significant. From engineering perspective, reaching movement that human perform while approaching an object for grasping can be modeled as a control system whose the the inputs are the motor commands from the central nervous system (CNS). To model the reach-to-grasp behavior in a specific grasping task, it is required to obtain the law that governs this behavior. In other words, what is the objective function that human attempts to optimize while approaching a certain object in an optimal way according to the principle of optimality [1]. Generally, objective function is the most concise representation of human behaviors [2].

In general, Inverse Reinforcement Learning (IRL) has been successfully used in the literature to model the *reward* function that best models a given human behavior [3]. Reward is the opposite to the cost term from machine learning perspective. IRL is in responsible for recovering the human reward/cost function given demonstrated sequence(s) of human actions. The original IRL Algorithm assumes that the reward is a parametric function that is a combination of weighted selected features. For instance, in [4] the IRL approach has been used to model the pedestrian behavior to let the robot imitate it. Furthermore in [5] a car driving simulator has been used to learn the different driving styles of the demonstrator using IRL technique. Using similar ideas, in [6] the IRL was used to learn different parking behaviors.

Inspired by the idea of the original IRL Algorithm and its successful applications, many researchers participate into further refinements of IRL such as [7] and [8]. One more refinement was to tackle the IRL problem from the optimal control perspective. Particularly, Inverse Optimal Control (IOC) has been proposed to infer the objective function that when applied in the forward optimal control mode it produces the best match for the demonstrated trajectory. For instance, in [9] the locomotion behavior of a human approaching a certain goal with different direction has modeled by means of solving the IOC problem. It was assumed that the objective function is a combination of several criteria that have to be minimized by the human (e.g. the traveled time, acceleration and the relative orientation between human and goal direction). Inverse Linear Quadratic Regulator (ILQR) has been introduced as a variant for the IOC approach. It models the human motor movement as a Linear Time Invariant (LTI) system while assuming a quadratic cost function that is minimized by the demonstrator. Given the dynamics of the movement under consideration, ILQR aims to solve the optimization problem that minimizes the error between both the estimated behavior and the given optimal or near-optimal demonstration.

Learning from a small amount of demonstrations is a challenge in machine learning [10]. The robustness of the learned cost function which models a given behavior is strongly

978-1-4673-9360-7/15/$31.00 2015 IEEE

affected by the number of demonstrated examples [11]. In literature, it has been often assumed that the training examples are available a priori and learning is done as a one-shot process. Despite the applicability of such a batch training approach, it is clear that, the learned cost function exclusively represents the available examples and there is no guarantee that it can be adapted with new demonstrations for the same behavior.

In this research we learn the cost function behind the reach-to-grasp behavior in the framework of ILQR. In contrast to the mentioned work in the literature, the proposed approach incorporates an evolutionary derivative free optimization technique; namely, Particle Swarm Optimization (PSO). Furthermore, we address the problem of incremental learning of such cost function where we aim to answer the question of how the already learned cost could be altered if another demonstrations are available even with different characteristic trajectory. To date, the incremental IOC has received scant attention in the research though its importance towards overcoming the problem of over-fitting [12].

The rest of this paper is organized as follow: In Section II, the forward LQR problem is briefly reviewed. The proposed Incremental IOC Algorithm is explained in details in Section III. Results for modeling the offered reach-to-grasp behavior are detailed and discussed in Section IV. Finally, conclusion is given in Section V with remarks on the future implications.

II. Linear Quadratic Regulator

The theory of optimal control aims to operate a dynamic system at minimum cost. The cost function is defined as a sum of key measurement deviations from their targets. LQ problem is a variant of optimal control in which the system dynamics are given by a group of linear differential equations and the performance index is a quadratic function. Solution for such LQ problem is provided by the linear-quadratic regulator (LQR), a controller that is responsible for finding the full state-feedback control law for a continuous-time linear system, which is summarized as follow:

Consider a dynamic system for which the state is described by the following linear state equations:

$$\dot{x} = Ax + Bu; \qquad x(0) = x_0 \tag{1}$$

$$y = Cx + Du \tag{2}$$

with a performance index J which has to be minimized and represented in the following quadratic form:

$$J = \int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t) + 2x(t)^T N u(t)\big) dt \tag{3}$$

where Q, R and N are the state, input and cross-coupling cost matrices respectively such that: $Q - NR^{-1}N^T \succeq 0$ and $R = R^T \succ 0$, where $\succeq$ and $\succ$ denotes semi-positive definite and positive definite matrices.

Given that (A,B) is controllable and (A,C) is detectable, the optimal control input which minimizes the cost function in Eq. (3) is given by:

$$u(t) = -Kx(t) \tag{4}$$

where

$$K = R^{-1}(B^T P + N^T) \tag{5}$$

in which P is obtained by solving the following Continuous-time Algebraic Riccati Equation (CARE):

$$A^T P + PA - (PB + N)R^{-1}(B^T P + N) + Q = 0 \tag{6}$$

The LQR problem in Eq. (1)-(6) is used in the forward mode, that is, given a set of weight matrices Q, R and N, the feedback control law K that minimizes the cost function given in Eq. (3) can be obtained. More details regarding LQR problem are given in [13]. In this work, the solution for the inverse problem of LQR is given more attention. Specifically, the cost function has to be recovered given the optimal reach-to-grasp behavior of a linear continuous-time system. This inverse problem is addressed in the following.

III. The Developed Incremental-ILQR

A. Problem Statement

The aim of solving the Inverse Optimal Control (IOC) problem is to retrieve the cost function that used to reproduce the measured data perfectly. Here, the IOC is applied for modeling biological behaviors, namely study the optimality criteria of human reach-to-grasp movements. This problem is formulated as follow:

The LQR problem is considered with an evaluation index

$$\min_{u(.)} \int_0^\infty \big(x(t)^T Q x(t) + u(t)^T R u(t) + 2x(t)^T N u(t)\big) dt \tag{7}$$

subject to:

$$\dot{x} = Ax + Bu \tag{8}$$

$$x(0) = x_0 \tag{9}$$

In this paper, retrieving the unknown objective function for the given dynamic model, Eq. (8), with given initial condition in Eq. (9) is the main objective. In the rest of this paper, it is assumed that the cross-coupling matrix N is zero since it is convenient to separate the state and the input terms of the objective function.

In inverse problem, the output response $y(t) \in \mathbb{R}^n$ is measured at equally time intervals t for a total L duration. Here, this measured response is assumed to be optimal since it is acquired from a human that follows the principle of optimality. It is required to obtain the exact weighting matrices Q and R that decrease the discrepancy between the estimated response $\bar{y}(t)$ and the measured response $y(t)$. Figure 1 illustrates the iterations done to find the best fit trajectory (in blue) that best matches the given optimal trajectory (in red). Selection of the quadratic cost function in Eq. (7) explains that matrix Q is responsible for penalizing the deviation of states from zero, while matrix R is in charge of penalizing the control effort $u(t)$ to be minimum. In formal terms, the problem for determining the weighting matrices Q and R could be represented as an optimization problem as follow:

$$\min_{Q,R} \sum_{t=0}^{L} \|\bar{y}(t; Q, R) - y(t_n)\|^2 \tag{10}$$

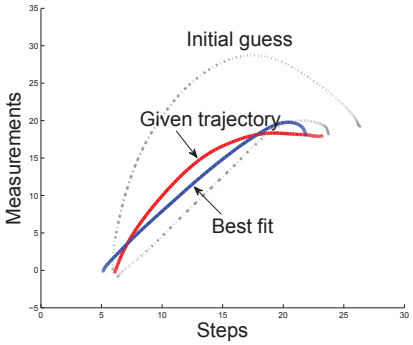


Fig. 1: Steps of obtaining the cost function that best matches a given measurements using inverse optimal control

where $\bar{y}(t;Q,R)$ is the simulated output response for the hypothesized $\bar{Q}$ and $\bar{R}$. It could be obtained by solving the forward problem of LQR and simulating the constructed closed loop system as follow:

$$\bar{y}(t;\bar{Q},\bar{R}) = Cx(t) \tag{11}$$

such that:

$$\dot{x} = (A - B\bar{K})x; \qquad x(0) = x_0 \tag{12}$$

$$\bar{K} = \bar{R}^{-1}B^T P \tag{13}$$

$$A^T P + PA - PB\bar{R}^{-1}B^T P + \bar{Q} = 0 \tag{14}$$

The optimization problem in Eq. (10) is solved using an existing stochastic method called Particle Swarm Optimization (PSO) which is quickly converge to a solution close to the optimal [14].

B. PSO-based ILQR Algorithm

The required matrix Q is formed as $Q = M^T M$ in such a way that ensures the stability condition, $Q > 0$ and $Q^T = Q$. Furthermore, matrix R which is concerned with the control effort is assumed to be a diagonal matrix, $R = diag[\beta_1 \ \beta_2 \ \dots \beta_{n_u}]$ with no loss of generality where n_u is the number of control inputs. In this formulation, the PSO is in charge of searching for the elements in M along with the factors β_i that minimize the discrepancy between both the measured and the hypothesized responses in Eq. (10). The average behavior, $y^*(t)$, will be considered if there exist more than one demonstration to learn the human behavior. This assumes that the all available demonstrations have the same initial state x_0 with same demonstration length, i.e. *samples/demonstration*.

There is an ambiguity problem for the given inverse LQR problem in Eq. (10); for example, if both matrices Q and R multiplied by a scalar $\lambda > 0$, the same output response will be obtained no matter what value of λ is. Consequently, it is expected to get different solutions for the inverse problem even if the same demonstration with same conditions exist. Therefore, an additional step is performed to obtain the min-max normalization for both matrices Q and R which are denoted by Q_n and R_n respectively. These normalization is obtained as follow:

$$Q_n = \frac{Q - min(Q)}{min(Q) - max(Q)} \tag{15}$$

$$R_n = \frac{R - min(R)}{min(R) - max(R)} \tag{16}$$

Algorithm 1 PSO-based Inverse LQR Algorithm

Input:
System dynamics: A, B, C, D and x_0.
N_d Human Demonstrations: $y_j(t)$, $j \in [1, N_d]$, $t \in [0, L]$.
Threshold of convergence: ϵ

Output:
The normalized behavior cost, Q_n, R_n

1: $y^*(t) = \frac{1}{N_d}\sum_{i=1}^{Nd} y_i(t_n)$ ▷ average behavoir is obtained
2: $p_0 \leftarrow random()$ ▷ inititalze inital particles with random values
3: $i \leftarrow 1$ ▷ iterations counter
4: **while** $SSE_i > \epsilon$ **do** ▷ no convergence achieved
5: $M_i \leftarrow p_i(1 : n^2_{states})$ ▷ extract M matrix
6: $\beta_i \leftarrow p_i(end - n_{control} : end)$ ▷ extract β scale
7: $\bar{Q}_i \leftarrow M^T M$ ▷ formulate Q_i matrix
8: $\bar{R}_i \leftarrow diag(\beta_{i1}\beta_{i2} \dots \beta_{in_u})$ ▷ formulate R_i matrix
9: $\bar{K}_i \leftarrow LQR(A, B, Q_i, R_i)$ ▷ solve forward LQR problem
10: $\dot{x} \leftarrow (A - B\bar{K}_i)x; x(0) = x_0$ ▷ closed loop state equation at instance i
11: $\bar{y}_i(\bar{Q}_i, \bar{R}_i, t) \leftarrow Cx$ ▷ response at instance i
12: $SSE_i \leftarrow \sum_{t=0}^{L} \left(\bar{y}_i(\bar{Q}_i, \bar{R}_i, t) - y^*(t)\right)^2$ ▷ fitness function
13: $i \leftarrow i + 1$ ▷ increment
14: $p_i \leftarrow$ **pso**() ▷ hypothesized solution at instance i
15: $Q_n \leftarrow Q_i - min(Q_i)/min(Q_i) - max(Q_i)$ ▷ normalize
16: $R_n \leftarrow R_i - min(R_i)/min(R_i) - max(R_i)$ ▷ normalize
17: **return** Q_n, R_n ▷ optimal normalized weighting matrices

Algorithm 1 explains the PSO-based Inverse LQR approach used to find the optimal normalized weighting matrices Q_n and R_n that formalize the objective function given N_d human demonstrations and system dynamics. Algorithm 1 starts with averaging the trajectories of the demonstrated behavior as shown in line 1. An initial guess for the PSO particles p is randomly assumed in line 2. Subsequently, at every instance i, the hypothesized weighting matrices $\bar{Q}_i$ and $\bar{R}_i$ are obtained from the candidate PSO solution p_i as depicted in lines 5:8. Henceforth, in line 9 the forward LQR problem is solved to find the estimated state feedback gain $\bar{K}_i$ by solving the CARE mentioned before. Lines 10 and 11 show how to find the estimated response $\bar{y}_i(\bar{Q}_i, \bar{R}_i)$ for the candidate weighting matrices. This is achieved by simulating the closed loop system dynamics with the given estimated feedback gain $\bar{K}_i$ and the initial condition x_0. Finally, the fitness of the estimated matrices is evaluated in line 12 in terms of the sum of squared errors (SSE) between both the hypothesized and the given average response. If this error is not below or equal to the specified convergence threshold ϵ, i.e. $SSE_i > \epsilon$, the algorithm will search again for another candidate solution.

Otherwise, it will be terminated since it converges to an optimal or a near-optimal solution. The off-the-shelf ***pso()*** function in line 14 is used to handle the PSO search. Once the optimal solution is found, the normalized weighting matrices Q_n and R_n could be easily calculated as shown in lines 14:15 using Eq. (15).

C. Incremental-ILQR Algorithm

Learning human behaviors from very few demonstrations causes an over-fitting problem where unexpected results are obtained for untrained demonstrations. It is nearly impossible to know how many demonstrations are required to learn the human behavior in all environment's circumstances. Besides, only a few and limited training examples could exist in some situations. This problem is addressed in this proposed approach by incrementally learn and refine the obtained cost function once new demonstrations are available. The same optimization criterion should be inferred for the same behavior with different circumstances [9]. It should Keep in mind that those new reach-to-grasp demonstrations might be in different circumstances than that in the batch learning scenario (e.g. a different object to be grasped, a different hand to grasp with, or even a different person who intends to grasp).

Suppose at a previous scenario $s-1$, the learned behavior cost matrices be $Q^{(s-1)}$ and $R^{(s-1)}$ which model the priori given demonstrations $y^{(s-1)}$ with an initial condition $x_0^{(s-1)}$. Subsequently, if another demonstration $y^{(s)}$ is available at the current learning scenario s, the so far learned weighting matrices should be altered to adapt with the new demonstration that has both different initial condition $x_0^{(s)}$ and trajectory length $L^{(s)}$. Algorithm 2 is developed to obtain the refined weighting matrices $Q^{(s)}$ and $R^{(s)}$. At first, the previously found optimal solution $p^{(s-1)}$ at the scenario $s-1$ is fed to the algorithm as an initial guess to obtain the estimated behavior $\bar{y}^{(s)}$ at the current scenario s. This makes the adaptation process significantly faster than running it from scratch. The key idea to achieve this adaptation is to refine the learned cost function in the same way as if this new unseen demonstration $y^{(s)}$ was exist with the a priori available demonstrations at the previous scenario $s-1$. Definitely, the cost function that best fits to the average trajectory y_s^* between the estimated $\bar{y}^{(s)}$ and the measured $y^{(s)}$ behavior trajectories at scenario s. This average trajectory is a weighed average between the estimated trajectory $\bar{y}^{(s)}$ obtained from the so far learned cost matrices at the previous scenario $s-1$ and the available demonstrated trajectory $y^{(s)}$ at the current scenario s. It is worth mentioning that the estimated trajectory is initially computed by simulating the system dynamics with the previous learned cost matrices $Q^{(s-1)}$ and $R^{(s-1)}$ while taking into account the new initial condition $x_0^{(s)}$ and the new trajectory length $L^{(s)}$. This step facilitates the computation of the average behavior that requires both trajectories to have the same initial condition and the same trajectory length. The obtained average behavior y_s^* is weighted by a confidence factor γ_s as follow:

$$y_s^* = (1-\gamma_s)\bar{y}^{(s)} + \gamma_s y^{(s)} \tag{17}$$

where $0 < \gamma_s \leq 1$ is the scenario confidence factor which is heuristically determined based on how the new demonstration is trusty in the current scenario s. In other words, in noisy scenarios, the acquired measurements are not sure and consequently small contribution of the currently demonstrated behavior $y^{(s)}$ will be accounted, i.e. small values for γ_s. On the other hand, more accurate measurements in less noise environments will contribute largely to the final cost function with a large value of γ_s. Since the PSO is in charge of minimizing the error between the average behavior y^* and the human demonstration at current scenario $y^{(s)}$, the fitness function at iteration i is specified in terms of SSE and with reference to Eq. (17) as follow:

$$SSE_i = \sum_{t=0}^{L} (\bar{y}^{(s)}(t) - y_s^*(t))^2 \tag{18}$$

$$= \gamma_s^2 \sum_{t=0}^{L} \left(\bar{y}^{(s)}(t) - y^{(s)}(t)\right)^2 \tag{19}$$

Hence, instead of computing the average behavior trajectory at each learning scenario, the fitness function in Algorithm 2 is multiplied by the square of the scenario confidence factor γ_s.

IV. Reach-to-Grasp behavior Modeling

To quantify the developed PSO-based Inverse LQR approach in behavior modeling, an experiment has been conducted to find the human response during a reach-to-grasp task. Modeling the point-to-point hand movements is an important problem in Human-Robot Interaction tasks in which the robot have to be aware of the human intention. The aim of this experiment is to find the cost function that human seeks to minimize while reaching a certain object to grasp it. Previous studies assume a heuristic cost function in this regard; such as minimizing the distance to an object or minimizing the deviation from the straight line towards the object. In [15], the predictive inverse optimal control method is used to estimate a probabilistic model of human motion in the reach-to-grasp behavior.

The instantaneous state $\vec{x}$ of the reaching task is represented by:

$$\vec{X}_t = [x_t\ y_t\ z_t\ \dot{x}_t\ \dot{y}_t\ \dot{z}_t\ \ddot{x}_t\ \ddot{y}_t\ \ddot{z}_t]^T$$

which contains the position, velocity, and acceleration vectors towards the target to be grasped. Velocities and accelerations are found according to difference equations up to a scale factor.

Reaching task can easily be expressed as a linear dynamics model with the control vector $\vec{u_t} = [\dddot{x}_t\ \dddot{y}_t\ \dddot{z}_t]^T$ representing the jerk; the time derivative of the hand acceleration and the measurements $\vec{Y_t} = [x_t\ y_t\ z_t]^T$ represent the position vector of the human hand. Under this dynamic model, reaching an object motion follows a linear relationship:

$$\dot{\vec{X}}_t = A\vec{X}_t + B\vec{u_t} \tag{20}$$

$$\vec{Y_t} = C\vec{X}_t \tag{21}$$

where A, B and C are the given model matrices.

Algorithm 2 Evolving PSO-based Inverse LQR Algorithm

Input:

Optimal solution at scenario $s-1$: $p^{(s-1)}$.
System dynamics: A, B, C, D.
Initial state at current scenario s: $x_0^{(s)}$
Current demonstration: $y^{(s)}(t)$, $t \in [0, L^{(s)}]$
Threshold of convergence: ϵ
Confidence factor at scenario s: γ_s

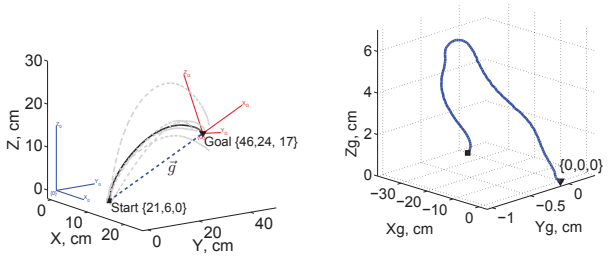
Output:

The normalized cost refined at scenario s: $Q_n^{(s)}$, $R_n^{(s)}$

$\{Q^{(s-1)}, R^{(s-1)}\} \leftarrow p^{(s-1)}$ ▷ calculate weighting matrices at scenario $s-1$
$K^{s-1} \leftarrow LQR(A, B, Q^{(s-1)}, R^{(s-1)})$ ▷ Gain at scenario $s-1$
$\dot{x} \leftarrow (A - BK^{s-1})x; x(0) = x_0^{(s)}$ ▷ closed loop state equation
$\bar{y}^{(s)} \leftarrow Cx$ ▷ estimated response at current scenario
$i \leftarrow 1$ ▷ iterations counter
while $SSE_i > \epsilon$ **do** ▷ no convergence achieved
  $M_i \leftarrow p_i(1 : n^2_{states})$ ▷ extract matrix M
  $\beta_i \leftarrow p_i(end - n_{control} : end)$ ▷ extract β scale
  $\bar{Q}_i \leftarrow M^T M$ ▷ formulate Q_i matrix
  $\bar{R}_i \leftarrow diag(\beta_{i1}\ \beta_{i2}\ \ldots \beta_{in_u})$ ▷ formulate R_i matrix
  $\bar{K}_i \leftarrow LQR(A, B, Q_i, R_i)$ ▷ solve forward LQR problem
  $\dot{x} \leftarrow (A - B\bar{K}_i)x; x(0) = x_0$ ▷ closed loop state equation at instance i
  $\bar{y}_i^{(s)}(\bar{Q}_i, \bar{R}_i, t) \leftarrow Cx$ ▷ response at instance i
  $SSE_i \leftarrow (1-\gamma_s)^2 \sum_{t=0}^{L} \left(\bar{y}_i^{(s)}(\bar{Q}_i, \bar{R}_i, t) - y^{(s)}(t)\right)^2$ ▷ fitness function
  $i \leftarrow i + 1$ ▷ increment
  $p_i \leftarrow$ **pso**() ▷ hypothesized solution at instance i
$Q_n^{(s)} \leftarrow Q_i - min(Q_i)/min(Q_i) - max(Q_i)$ ▷ normalize
$R_n^{(s)} \leftarrow R_i - min(R_i)/min(R_i) - max(R_i)$ ▷ normalize
return $Q_n^{(s)}, R_n^{(s)}$ ▷ optimal normalized weighting matrices at the new secenario s.

A. Experimental Setup

The Human Grasping datasets performed at the Lab. of Heiner Deubel [16] is used here in order to analyze and quantify our developed PSO-based Inverse LQR approach. A Polhemus Liberty system with six magnetic sensors was used for recording the data. A sensor was attached to each fingertip, positioned on the fingernail and one was placed on the dorsum of the hand. Five subjects were asked to perform the 31 grasps for different objects. Initially, the human hand was placed in front of the table in a flat hand posture. Upon a starting signal, the subject grasped an object, lifted the object, put it down again and retreated the hand to the starting position. The data acquiring starts when the hand begins to move and ended when the hand was returned to the initial position. Two grasps have performed for each object, the first one was used to create the training data and the second one for the test data.



(a) (b)

Fig. 2: (a) Averaged behaviour over five subjects for the first scenario: grasping a tennis ball. (b) Transformed behaviour trajectory in the goal-dependent axis system.

B. Preprocessing Hand Movements

The used datasets have to be preprocessed before have involved in the leaning step to fit the aim of our study. Firstly, the hand dorsum position $\vec{Y}_t = [x_t\ y_t\ z_t]^T$ is extracted along only the reaching step. Here, this position vector is initially expressed in a global reference frame defined by the original datasets. Subsequently, for each given scenario of hand motion, the average stereotypical behavior trajectory over five subjects is obtained assuming that all demonstrations per scenario start from the same position and end up with the goal to be grasped as depicted in Figure 2a. Particularly, four scenarios are involved in the experiment where each scenario is characterized by grasping a certain object (tennis ball, CD, credit card and coin) at different location. Finally, the average dorsum motion trajectory has projected to a new goal-dependent axis system in which the control action is invoked to direct the motion to the origin. The X_g-axis of the new axis system is aligned with the start-to-gaol direction while both the Y_g and Z_g axes are orthogonal as shown in Figure 2b. Reaching trajectories are primarily aligned along the $\vec{g}$ vector between the starting and the goal points. Therefore, the goal-dependent transformation facilitates the cost learning process since the reaching movement has inconsiderable deviation along the lateral Y_g-axis. Goal-dependent transformation is obtained via a homogeneous transformation T which is composed of a 4×4 rotation matrix R_g followed by a 4×4 translation matrix $T_g(g_x, g_y, g_z)$. R_g is defined as the rotation matrix that rotates the original x-axis to be in alignment with the vector towards the goal $\vec{g}$ as shown in Figure 2a. Accordingly, T_g is the translation matrix that translates the origin to the goal location $g = [g_x\ g_y\ g_z]^T$.

$$T = (T_g * R_g) \tag{22}$$

$$T_g = Trans(g_x, g_y, g_z) \tag{23}$$

$$R_g = R_{k,\theta} \tag{24}$$

where $R_{k,\theta}$ is called the axis/angle of rotation R_g which is obtained using MATLAB function *vrrotvec* as follow:

$$R_{k,\theta} = vrrotvec(\vec{i}, \vec{g}) \tag{25}$$

C. Reaching movement cost function

The PSO-based Inverse LQR algorithm 1 is applied for each of the four grasping scenarios given the preprocessed demonstrations explained above. The input parameters for the

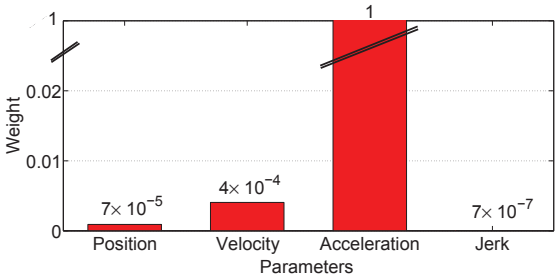


Fig. 3: Contribution of each parameters in cost function of reach-to-grasp movement (normalized by the maximum value).

algorithm are chosen as follow: five human demonstrations per scenario ($N_d = 5$), samples per demonstration ($L = 140$) and a convergence threshold ($\epsilon = 2$). The contribution of each state (position, speed and acceleration) besides the control effort (jerk) is measured after obtaining the composed matrix χ that combines both the Q and R matrices as follow:

$$\chi = \begin{bmatrix} Q & \mathbf{0}_{9\times 3} \\ \mathbf{0}_{3\times 9} & R \end{bmatrix} \tag{26}$$

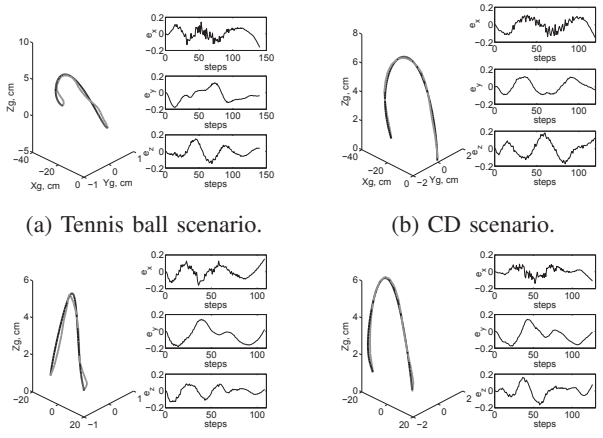
The main diagonal of the composed matrix χ indicates the normalized weight of each element for the total cost function that represents the reaching behavior. As depicted from Figure 3, the total learned cost is mainly composed of position, velocity, acceleration and jerk parameters with different contribution weights. It is worth pointing out that the contribution of the hand acceleration is significant compared to the other factors composing the reaching cost. This result is consistent with the kinematic model of reaching found in neuroscience literature [17] and [18]. They suggest that the human reach an object with maximum motion *smoothness* which is defined as having small high-order derivatives that is the squared acceleration in this study. Figure 4 shows the results of applying PSO-based Inverse LQR approach for the four grasping scenarios. The solid black line in all sub-figures represents the respective computed optimal trajectory for the identified set of weighting matrices computed by solving the optimal forward problem. Gray trajectory denotes the average measured demonstrations for each scenario used as bases for the computation. The discrepancies between modeled and measured behaviors in all cases are obtained in terms of the error between them in each axis: e_x, e_y and e_z as shown in the sub-figures for each scenario.

D. Incremental cost learning

To quantify the Evolving PSO-based Inverse LQR developed in Algorithm 2, the obtained cost function is incrementally refined with the given successive scenarios. At each given scenario s, the Frobenius Norm $FN^{(s)}$ of the difference between two successive normalized learned composite cost matrices, $\Delta\chi_s = \bar{\chi}^{(s)} - \bar{\chi}^{(s-1)}$, is measured. This indicates how the learned cost function generalizes for unseen scenarios for the behavior under study. This measure is given as follow:

$$FN^{(s)} = \|\Delta\chi_s\|_F \tag{27}$$

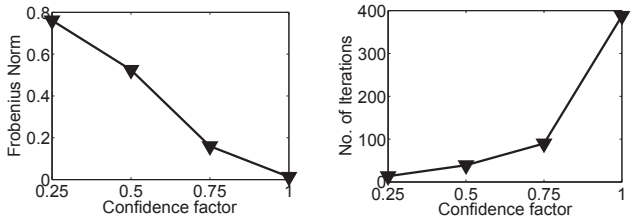
$$= \sqrt{trace(\Delta\chi_s^T \Delta\chi_s)} \tag{28}$$



(a) Tennis ball scenario. (b) CD scenario.

(c) Credit Card scenario. (d) Coin scenario.

Fig. 4: Results of PSO-based Inverse LQR performed for the four scenarios. The left plot in each sub-figure presents the 3D fit between the measurements (dashed line) with the estimated optimal trajectory (solid line) produced by the identified objective function. The other plots in the right of each sub-figure are the 2D projection in the x-y and y-z plane of the fit.



(a) (b)

Fig. 5: Results of incremental cost learning with four different confidence factors γ_s. (a) Frobenius Norm between the composite cost matrices of successive scenarios. (b) the number of iterations required to best fit the measured scenario.

where $trace(A)$ gives the trace of a given matrix A. In addition, the number of iterations required to best acquire the new demonstration is calculated at each scenario.

At first, the algorithm learns the composite cost matrix $\chi^{(1)}$ from the scenario of one subject grasping a Compact Disk (CD). Subsequently, here six test scenarios are used to incrementally refine the obtained composite cost matrix. These test scenarios are differ in the starting points, number of subjects, objects to be grasped and length of demonstrations. These scenario are selected to incrementally refine the learned composite cost function as follow:

- scenario 1: One subject grasping a CD.
- scenario 2: Five subjects grasping a tennis ball.
- scenario 3: Three subjects grasping a CD.
- scenario 4: Five subjects grasping a Credit Card.
- scenario 5: Five subjects grasping a coin.
- scenario 6: One subject grasping a plate.

Figure 5 shows the average Frobenius Norm and the average number of iterations required after applying the developed Evolving PSO-based Inverse LQR approach with different confidence factor $\gamma_s = [0.25, 0.5, 0.75, 1]$. Figure 5a indicates

a reduction in the difference between two successive modeled cost functions of the reaching behavior at confidence factor of $\gamma_s = 1$ which represents fully trusted scenario. It also implies that the learned cost function is adaptively generalized for the unseen demonstrations learned incrementally. However, the average number of iterations required to fit the measured behavior is increasing with the confidence factors. Confidence of $\gamma_s = 0.75$ could be the best choice for the current available dataset since this confidence values gives small Frobenius Norm while having small number of iterations as well.

V. Conclusion

A novel PSO-based Inverse Linear Quadratic Regulator has been developed for learning the optimization criteria underlying a given human reach-to-grasp movements. The developed algorithm is based on using one of the evolutionary meta-heuristic optimization techniques, namely Particle Swarm Optimization. In addition, an incremental inverse LQR Algorithm has been developed to incrementally refine the cost function when receiving untrained demonstrations even with different initial condition and length. The developed approach has been confirmed through retrieving the cost function behind the reach-to-grasp motor movements. Several objects were used in different situations from an available grasping dataset. The obtained cost function is proven to be consistent with the neuroscience literature of the human hand reaching behavior. Additionally, in this application, as well as, the incremental learning of the cost function proves reduction in the total time required for learning human behavior while provides a step toward the cost function generalization to overcome the overfitting problem.

References

[1] E. Todorov, "Optimality principles in sensorimotor control," *Nature neuroscience*, vol. 7, no. 9, pp. 907–915, 2004.

[2] A. Y. Ng, S. J. Russell *et al.*, "Algorithms for inverse reinforcement learning." in *Icml*, 2000, pp. 663–670.

[3] M. C. Priess, R. Conway, J. Choi, J. M. Popovich, and C. Radcliffe, "Solutions to the inverse lqr problem with application to biological systems analysis," *Control Systems Technology, IEEE Transactions on*, vol. 23, no. 2, pp. 770–777, 2015.

[4] S.-Y. Chung and H.-P. Huang, "A mobile robot that understands pedestrian spatial behaviors," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 5861–5866.

[5] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 1.

[6] P. Abbeel, D. Dolgov, A. Y. Ng, and S. Thrun, "Apprenticeship learning for motion planning with application to parking lot navigation," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 1083–1090.

[7] D. Ramachandran and E. Amir, "Bayesian inverse reinforcement learning," *Urbana*, vol. 51, p. 61801, 2007.

[8] A. Boularias, J. Kober, and J. R. Peters, "Relative entropy inverse reinforcement learning," in *International Conference on Artificial Intelligence and Statistics*, 2011, pp. 182–189.

[9] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotionan inverse optimal control approach," *Autonomous robots*, vol. 28, no. 3, pp. 369–383, 2010.

[10] B. D. Ziebart, "Modeling purposeful adaptive behavior with the principle of maximum causal entropy," 2010.

[11] C. Giraud-Carrier, "A note on the utility of incremental learning," *AI Commun.*, vol. 13, no. 4, pp. 215–223, Dec. 2000. [Online]. Available: http://dl.acm.org/citation.cfm?id=1216442.1216444

[12] S. J. Lee and Z. Popović, "Learning behavior styles with inverse reinforcement learning," in *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4. ACM, 2010, p. 122.

[13] H. Kwakernaak and R. Sivan, *Linear optimal control systems*. Wiley-interscience New York, 1972, vol. 1.

[14] J. Kennedy, J. F. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm intelligence*. Morgan Kaufmann, 2001.

[15] M. Monfort, A. Liu, and B. Ziebart, "Intent prediction and trajectory forecasting via predictive inverse linear-quadratic regulation," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[16] T. Feix, J. Romero, C. H. Ek, H. Schmiedmayer, and D. Kragic, "A Metric for Comparing the Anthropomorphic Motion Capability of Artificial Hands," *Robotics, IEEE Transactions on*, vol. 29, no. 1, pp. 82–93, Feb. 2013. [Online]. Available: http://grasp.xief.net

[17] S. Ben-Itzhak and A. Karniel, "Minimum acceleration criterion with constraints implies bang-bang control as an underlying principle for optimal trajectories of arm reaching movements," *Neural Computation*, vol. 20, no. 3, pp. 779–812, 2008.

[18] B. Ziebart, A. Dey, and J. A. Bagnell, "Probabilistic pointing target prediction via inverse optimal control," in *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*. ACM, 2012, pp. 1–10.