Engineering Applications of Artificial Intelligence 50 (2016) 115–124

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Adaptive learning of human motor behaviors: An evolving inverse optimal control approach

Haitham El-Hussieny [a,1,*], A.A. Abouelsoud [a,2], Samy F.M. Assal [a,3], Said M. Megahed [b]

[a] Mechatronics and Robotics Engineering Department, School of Innovative Design Engineering, Egypt-Japan University of Science and Technology (E-JUST), Egypt

[b] Mechanical Design and Production Engineering Department, Faculty of Engineering, Cairo University, Egypt

ARTICLE INFO

Article history:
Received 8 September 2015
Received in revised form
6 January 2016
Accepted 8 January 2016
Available online 2 February 2016

Keywords:
Inverse optimal control
Linear Quadratic Regulator
Behavior modeling
Particle Swarm Optimization
Incremental learning

ABSTRACT

Understanding human behaviors has received considerable attention in neuroscience literature. Existing research addressed the main question: given measurements of a certain human movement, what is the underlying optimality criteria that human has optimized to fulfill this movement? Inverse Optimal Control (IOC) is a well-established approach to understand the biological movements in terms of the optimal control theory. IOC learns the criterion that best describes the demonstrated human behavior. Thus far, gradient-based techniques have been used to obtain the unknown behavior cost. However, these techniques are limited by locating only local optimum parameters. In this paper, behavior learning is modeled as an Inverse Linear Quadratic Regulator (ILQR) problem, where linear behavior dynamics and a quadratic cost are assumed. An efficient meta-heuristic technique, Particle Swarm Optimization (PSO), is used to retrieve the unknown cost in the proposed ILQR problem. Moreover, an evolving-ILQR algorithm is proposed to refine the learned cost once new unseen demonstrations exist to overcome the over-fitting problem. The reach-to-grasp behavior is studied to quantify the proposed approaches. Results are encouraging and show consistency with that in neuroscience literature. Meanwhile, the evolving-ILQR algorithm has quantified in successive scenarios, where the so far retrieved behavior cost has incrementally refined once new unseen demonstrations are available.

1. Introduction

Modeling the daily human motor activities in certain contexts (i.e. grasping objects, driving a vehicle or even playing tennis) has become a central issue for human centered computation. There is a growing body of literature that recognizes how the understanding of human motor behaviors is significant. Particularly, in Human–Robot Interaction (HRI) and Human–Computer Interaction (HCI) applications, understating how human behaves in a given context allows for an accompanying agent to recognize the human motions beforehand (i.e. intent prediction) to facilitate the human guided assistance (Khokar et al., 2013; Ahmad et al., 2015). Moreover, in bio-mechanics research, the formulation of motor behaviors plays a critical role in the assessment of the dissimilarities that could exist between healthy and disordered subjects (Abaid et al., 2012). Additionally, in *learning by demonstration* the retrieved model for the human behavior could be used to teach an agent how to act in a natural human-like way to fulfill a certain task (Mombaur et al., 2010a, 2013).

From engineering perspective, biological movement can be modeled as a system whose motor commands are the inputs from the central nervous system (CNS) controller. In particular, human sensorimotor controls are assumed to be done in an optimal way following the principle of optimality (Todorov, 2004). To model a certain human behavior in a specific 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 achieving that behavior. Objective function is the most concise representation of human behaviors (Ng et al., 2000). From the optimal control theory, the human behavior objective function could be recovered by the means of applying the Inverse Optimal Control (IOC) framework (Zhifei and Joo, 2012). IOC is concerned with finding the cost function that best explains the performance criteria of a demonstrated behavior. Once a dynamic model of the behavior under consideration is available, the IOC problem could be solved to recover the cost function from the demonstrations of this behavior.

* Corresponding author.
E-mail address: haitham.elhussieny@ejust.edu.eg (H. El-Hussieny).
[1] On leave: Electrical Engineering Department, Shoubra Faculty of Engineering, Benha University, Egypt.
[2] On leave: Electronics and Communications Engineering Department, Faculty of Engineering, Cairo University, Egypt.
[3] On leave: Department of Production Engineering and Mechanical Design, Faculty of Engineering, Tanta University, Egypt.

http://dx.doi.org/10.1016/j.engappai.2016.01.024
0952-1976/

Learning from a small amount of demonstrations is a challenge in machine learning in general and in IOC in particular (Ziebart). The robustness of the learned cost function which models a given behavior is strongly affected by the number of demonstrated examples (Giraud-Carrier, 2000). In the literature, it has often been 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. A wide range of situations requires learning the cost function in an incremental/recursive fashion. This will meet both the self-adaptation and the performance improvement capabilities which are prerequisite for machine learning algorithms (Newell and Simon, 1976). Meanwhile, systems with low memory could use such a learning approach to incrementally gather knowledge once new data becomes available with no need for memorizing all the training examples. To the best of our knowledge, development of a self-adapted behavior modeling system has not been attempted.

In the literature, several works have studied the problem of identifying the objective function behind a particular human motor movement. For instance, in Nakazawa et al. (2002) and Suleiman et al. (2008), this problem is addressed from the imitation point of view, i.e. the reproduction of the learned demonstrations over a humanoid robot with specific kinematic and dynamic ranges. Learning approaches were used to identify the parameters that best imitate the observed behavior within a given scenario without taking in mind what is the underlying optimality criteria behind such behavior. Unsatisfactory results were obtained when trying to reproduce the same behavior within unstructured dynamic environment (Atkeson and Schaal, 1997).

Other studies, such as Dragan and Srinivasa (2013) and El-Hussieny et al. (2015) assumed a heuristic cost function that human aims to minimize while performing certain behavior. This cost function was selected by intuition as a single feature that covers the human behavior. For instance, such heuristic cost function could be minimizing the distance to an intended location to model the human locomotion behavior. This technique is beneficial in scenarios where the learning from a priori given example is not allowed, such as in unknown and unpredictable environments. In this regard, a heuristic cost function combined of more than one feature is intractable due to the lack of a strategy that determines the weight of each term's contribution to this function.

Inverse Reinforcement Learning (IRL) has been successfully used in the literature to retrieve the *reward* function that best model a given behavior. Reward is the opposite to the cost term from machine learning perspective. IRL is in charge of 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 Chung and Huang (2010) the IRL approach was used to model the pedestrian behavior to let the robot imitate it. Furthermore, in Abbeel and Ng (2004) a car driving simulator was used to learn the different driving styles of the demonstrator using IRL technique. Using similar ideas, in Abbeel et al. (2008) the IRL was used to learn different parking behaviors.

Inspired by the idea of the original IRL algorithm and its successful applications, many researchers have participated into further refinements of IRL such as in Ramachandran and Amir (2007) and Boularias et al. (2011). One more refinement was to tackle the IRL problem from the optimal control perspective. Particularly, IOC has been proposed to infer the objective function that, when applied in the forward optimal control mode, produces the best match for the demonstrated trajectory. For instance, in Mombaur et al. (2010b) the locomotion behavior of a human approaching a certain goal with different directions was 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 directions).

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 assumes that a quadratic cost function is the function to be minimized by the demonstrator. Given the dynamics of the movement under consideration, ILQR aims to solve the optimization problem that minimizes the error between the estimated behavior and the given optimal or near-optimal demonstration. This estimated behavior is obtained as a result of solving the forward LQR problem with the cost function. For instance, an assumed quadratic cost function was obtained in Ziebart et al. (2012), which models the probabilistic dynamics of a 2D target pointing task. These pointing movements were demonstrated by a human moving the mouse cursor toward the target of attention. Extension for modeling 3D human arm movement was introduced in Monfort et al. (2015). A recent work that models the human behavior using ILQR was proposed in Priess et al. (2015). It divides the problem of behavior modeling into two subsequent processes. First, the value of the control gain that best describes the sampled behavior was obtained using the least square minimization. Afterwards, two classical optimization techniques were proposed to find the cost function that best produces the estimated control gain. In this work, although the accumulated error may occur due to the two subsequent learning steps, the proposed approach gave satisfactory results when applied for modeling the stabilization behavior of a human seated on a moving robot.

With recent achievements in computational power and improvements in optimization algorithms, we badly need to investigate the improvements of using one of the meta-heuristic optimization techniques to learn the cost function for a given human behavior. Meta-heuristic provides a sufficiently good solution to an optimization problem, especially with incomplete or imperfect information (Blum and Roli, 2003). In this research, the cost function behind a demonstrated behavior is learned 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, the problem of incremental learning of such cost function is addressed in which, the way of the already learned cost can be altered is illustrated; especially if other demonstrations with different characteristics are available. This way mimics the natural way of human being in acquiring knowledge over time. Once new information becomes available, the learned knowledge is revised (i.e. evolve) to generalize for the new available information. To date, the incremental IOC has received scant attention in the research in spite of its importance towards overcoming the problem of over-fitting (Lee and Popović, 2010).

To illustrate how to find the cost function for behavior modeling with the developed evolving IOC approach, the reach to grasp movement is taken as an example for this illustration, in which human subject is asked to reach a certain object to grasp it. The developed evolving IOC framework is applied to obtain the cost function to be minimized, which human used while reaching an object. To strengthen the obtained results, the direct optimal control problem is then solved to compare the given and the simulated behaviors. Moreover, the adaptability of the algorithm in retrieving the human behavior is tested with new demonstrations that have never seen before. The results of the proposed approach are encouraging and show that the retrieved cost function is consistent with that in neuroscience literature. Additionally,

it is shown that the learned cost function could be recursively adapted for the novel demonstrations related to the behavior under discussion with a reduction in the learning time.

The rest of this paper is organized as follows: in Section 2, the forward LQR problem is briefly reviewed. The proposed evolving IOC algorithm is explained in detail in Section 3. Results for modeling the offered reach to grasp behavior are presented and discussed in Section 4. Finally, conclusion is given in Section 5 with remarks on the future implications.

2. Review of 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. Linear Quadratic (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 LQR, a controller that is responsible for finding the full state-feedback control law for a continuous-time linear system, which is summarized as follows.

Consider the dynamic system whose state $\boldsymbol{x}$ is described by the following linear state equations:

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

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

where A, B, C and D are the system matrices while u and y are the control and the output responses respectively; additionally with the following performance index J which has to be minimized:

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

where Q, R and N are the state, input and cross-coupling cost matrices respectively such that $Q - NR^{-1}N^T \succcurlyeq 0$ and $R = R^T \succ 0$, where $\succcurlyeq$ 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 (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 (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 (3) can be obtained. More details regarding LQR problem are given in Kwakernaak and Sivan (1972). In this paper, the solution for the inverse problem of LQR is given more attention. Specifically, the cost function has to be recovered given the optimal behavior of a linear continuous-time system. This inverse problem is addressed as follows.

3. The developed evolving-ILQR

3.1. Problem statement

The aim of solving the IOC problem is to retrieve the cost function given by (7) which models the human behavior under consideration. Specifically, the IOC is in charge of finding the Q, R and N matrices that reproduce the measured behavior perfectly. Here, the IOC is applied for modeling biological behaviors, namely study the optimality criteria of human movements. This problem is formulated as follows.

The LQR problem is considered with an evaluation index

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

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 (8) with given initial condition (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 the inverse problem, the output response $y(t) \in \mathbb{R}^n$ is measured at equal time interval t for a total L duration, where n denotes the number of obtained measurements. 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 $y(t)$. For instance, for a certain behavior of a human, Fig. 1 illustrates the generated trajectories (in blue) by the IOC, which is used to find the optimal trajectory $\bar{y}(t)$ (in green) that best matches the trajectory of the given measurement data $y(t)$ (in red) representing this certain behavior of a human. Note that the generated trajectories by the IOC are the trajectories that generated every iteration starting from the initial guess one until reaching the optimal trajectory $\bar{y}(t)$. A certain optimization stopping criterion is used to identify this optimal trajectory such as the minimum allowable error between the guess and the measured trajectories is met. Selection of the quadratic cost function in (7) explains that the matrix Q is responsible for penalizing the deviation of states from zero, while the 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 follows:

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

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

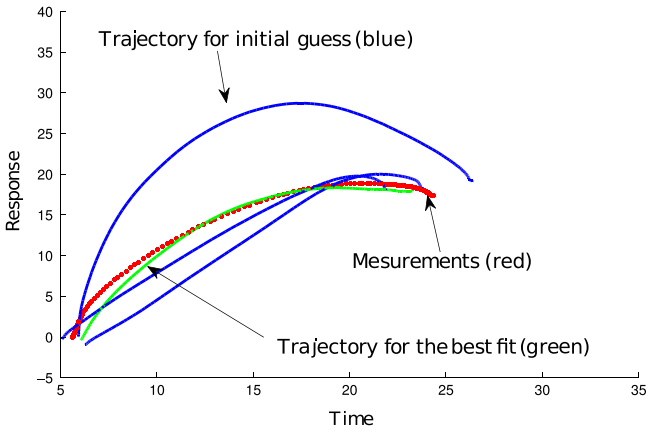

Fig. 1. Generation of the initial trajectories (blue) using IOC to find the optimal trajectory (green) that best matches the trajectory of the given measurement data (red). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

problem of LQR and it simulates the constructed closed loop system as follows:

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

such that:

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

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

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

The optimization problem in (10) is solved using an existing stochastic method called Particle Swarm Optimization (PSO) which is quickly converged to a solution close to the optimal (Kennedy et al., 2001).

3.2. Particle swarm optimization

To solve the problem given in (10), the PSO technique is used. PSO is a computational algorithm that is recursively trying to find the best candidate solution with respect to a given fitness function. At the beginning, a random population of candidate solutions, called particles, are spanned over the solution search space. Particles change their place and speed according to the best local positions they have achieved so far. This local position called *pbest*. Also, the best global position overall the particles, *gbest*, influences the movement of the whole swarm. Particles will update their positions searching for optimal places imitating the social behavior of bird flocking and fish schooling. This search will continue until satisfying one of the stopping criteria such as reaching either the maximum number of iterations or the maximum run time, the objective function reaches a specified limit or no change exist in the objective function.

A particle p within the swarm S updates its current position $x_p^{(i)}$ and velocity $v_p^{(i)}$ at instance i according to the two best locations obtained, $pbest_p^{(i)}$ and $gbest^{(i)}$ using the following equations:

$$v_p^{(i)} = v_p^{(i-1)} + c_1 r_1 (pbest_p^{(i)} - x_p^{(i)}) + c_2 r_2 (gbest^{(i)} - x_p^{(i)}) \tag{15}$$

$$x_p^{(i+1)} = x_p^{(i)} + v_p^{(i)} \tag{16}$$

where r_1 and r_2 are random numbers between (0,1). c_1 is called the *cognitive component* while c_2 is called the *social component* which should satisfy the condition $c_1 + c_2 > 4$. In the literature, it is common to have $c_1 = c_2 = 2$.

3.3. 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, the matrix R which is concerned with the control effort is assumed to be a diagonal matrix, $R = diag[\beta_1\ \beta_2\ \ldots \beta_{n_u}]$ 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 (10). The average behavior, $y^*(t)$, will be considered if there exist more than one demonstration to learn the human behavior. This assumes that all the available demonstrations have the same initial state x_0 with the same demonstration length, i.e. *samples/demonstration*.

There is an ambiguity problem for the given ILQR problem in (10); for example, if both matrices Q and R are 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. This normalization is obtained as follows:

$$Q_n = \frac{Q - \min(Q)}{\min(Q) - \max(Q)} \tag{17}$$

$$R_n = \frac{R - \min(R)}{\min(R) - \max(R)} \tag{18}$$

This normalization is acceptable since in behavior modeling, finding the relation between the contribution of each state or input related to each others in the overall cost function is of interest. Algorithm 1 explains the PSO-based ILQR 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. The algorithm 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 $\overline{Q_i}$ and $\overline{R_i}$ 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 $\overline{K_i}$ by solving the CARE mentioned before. Lines 10 and 11 show how to find the estimated response $\overline{y_i}(\overline{Q_i}, \overline{R_i})$ for the candidate weighting matrices. This is achieved by simulating the closed loop system dynamics with the given estimated feedback gain $\overline{K_i}$ 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 responses. 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. This search for optimal parameters that constitute the cost function is handled by the PSO technique. 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 and 15 using (17). These matrices form the cost function that human operator seeks to minimize while demonstrating the given behavior. The obtained weighting matrices emphasize the weight of each state contribution in the total cost function for a demonstrated human behavior under consideration.

Algorithm 1. PSO-based ILQR 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 behavior is obtained
2: $p_0 \leftarrow random()$ ▷ initialize initial particles with random values
3: $i \leftarrow 1$ ▷ iterations counter
4: **while** $SSE_i > \epsilon$ **do** ▷ no convergence achieved
5: $M_i \leftarrow p_i(1 : n_{states}^2)$ ▷ extract M matrix
6: $\beta_i \leftarrow p_i(end - n_{control} : end)$ ▷ extract β scale
7: $\overline{Q_i} \leftarrow M^T M$ ▷ formulate Q_i matrix
8: $\overline{R_i} \leftarrow diag(\beta_{i1}\beta_{i2}\ldots\beta_{in_u})$ ▷ formulate R_i matrix
9: $\overline{K_i} \leftarrow LQR(A, B, Q_i, R_i)$ ▷ solve forward LQR problem
10: $\dot{x} \leftarrow (A - B\overline{K_i})x; x(0) = x_0$ ▷ closed loop state equation at instance i

11: $\overline{y_i}(\overline{Q_i}, \overline{R_i}, t) \leftarrow Cx$ ▷ response at instance i
12: $SSE_i \leftarrow \sum_{t=0}^{L} \left(\overline{y_i}(\overline{Q_i}, \overline{R_i}, t) - y^*(t)\right)^2$ ▷ 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

3.4. Evolving-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 incremental learning and refining the obtained cost function once new demonstrations are available. Since the biological motion exhibits invariant features (Soechting and Lacquaniti, 1981), the learned cost function should further generalize within the human optimality bounds with each new unseen demonstration. The same optimization criterion should be inferred for the same behavior with different circumstances (Mombaur et al., 2010b). For example, suppose that a single demonstration exists for learning the human behavior of grasping an object. The aforementioned PSO-based ILQR could be used to learn the cost function that should be minimized to meet this grasping behavior from the priori available batch demonstrations. After that, the obtained cost function has to be refined with the existence of new grasping demonstrations in subsequent scenarios. It should keep in mind that those new 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 are $Q^{(s-1)}$ 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 different initial conditions $x_0^{(s)}$ 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 $\overline{y}^{(s)}$ 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 existed with the 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 $\overline{y}^{(s)}$ and the measured $y^{(s)}$ behavior trajectories at scenario s should be retrieved. This average trajectory is a weighed average between the estimated trajectory $\overline{y}^{(s)}$ 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 follows:

$$y_s^* = (1-\gamma_s)\overline{y}^{(s)} + \gamma_s y^{(s)} \tag{19}$$

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 (19) as follows:

$$SSE_i = \sum_{t=0}^{L} (\overline{y}^{(s)}(t) - y_s^*(t))^2 \tag{20}$$

$$SSE_i = \gamma_s^2 \sum_{t=0}^{L} \left(\overline{y}^{(s)}(t) - y^{(s)}(t)\right)^2 \tag{21}$$

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. At noisy scenarios, low confidence factor exists and therefore the fitness function approaches the stopping criteria faster which results in more tendency toward the previous learned cost function. On the contrary, large value of confidence factor in trusted scenarios will yield more tendency toward learning a new cost function that significantly depends on the recent demonstrations. Although the Evolving-ILQR algorithm is run in an online fashion to adapt the retrieved cost matrices, the parameters of the PSO technique are tuned only once off-line at the beginning of the experiment. Hence, there is no possibility to change the tuned parameters of the PSO in real time. Therefore, this way of adaptation cannot damage the system due to the existence of non-suitable tuned parameters.

Algorithm 2. Evolving PSO-based ILQR 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 human demonstration at scenario s: $y^{(s)}(t)$, $t \in [0, L^{(s)}]$
Threshold of convergence: ϵ
Confidence factor at scenario s: γ_s

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

1: $\{Q^{(s-1)}, R^{(s-1)}\} \leftarrow p^{(s-1)}$ ▷ calculate weighting matrices at scenario $s-1$
2: $K^{s-1} \leftarrow LQR(A, B, Q^{(s-1)}, R^{(s-1)})$ ▷ Gain at scenario $s-1$
3: $\dot{x} \leftarrow (A - BK^{s-1})x; x(0) = x_0^{(s)}$ ▷ closed loop state equation
4: $\overline{y}^{(s)} \leftarrow Cx$ ▷ estimated response at current scenario
5: $i \leftarrow 1$ ▷ iterations counter
6: **while** $SSE_i > \epsilon$ **do** ▷ no convergence achieved
7: $M_i \leftarrow p_i(1 : n_{states}^2)$ ▷ extract matrix M
8: $\beta_i \leftarrow p_i(end - n_{control} : end)$ ▷ extract β scale
9: $\overline{Q_i} \leftarrow M^T M$ ▷ formulate Q_i matrix
10: $\overline{R_i} \leftarrow diag(\beta_{i1}\ \beta_{i2}\ \ldots \beta_{in_u})$ ▷ formulate R_i matrix

11: $\overline{K_i} \leftarrow LQR(A, B, Q_i, R_i)$ ▷ solve forward LQR problem
12: $\dot{x} \leftarrow (A - B\overline{K_i})x; x(0) = x_0$ ▷ closed loop state equation at instance i
13: $\overline{y}_i^{(s)}(\overline{Q}_i, \overline{R}_i, t) \leftarrow Cx$ ▷ response at instance i
14: $SSE_i \leftarrow (1-\gamma_s)^2 \sum_{t=0}^{L} \left(\overline{y}_i^{(s)}(\overline{Q}_i, \overline{R}_i, t) - y^{(s)}(t)\right)^2$ ▷ fitness function
15: $i \leftarrow i+1$ ▷ increment
16: $p_i \leftarrow$ **pso**() ▷ hypothesized solution at instance i
17: $Q_n^{(s)} \leftarrow Q_i - min(Q_i)/min(Q_i) - max(Q_i)$ ▷ normalize
18: $R_n^{(s)} \leftarrow R_i - min(R_i)/min(R_i) - max(R_i)$ ▷ normalize
19: **return** $Q_n^{(s)}, R_n^{(s)}$ ▷ optimal normalized weighting matrices at the new scenario s.

4. Reach-to-grasp behavior modeling

To quantify the developed PSO-based ILQR 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 HRI 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. Previous studies assume a heuristic cost function in this regard; such as minimizing either the distance to an object or the deviation from the straight line towards the object. In Monfort et al. (2015), the predictive IOC method is used to estimate a probabilistic model of human motion in the reach-to-grasp behavior. In contrast to the aforementioned approaches, the cost function that describes the reach-to-grasp task is required to be obtained using PSO which is an evolutionary and a derivative free optimization approach. This cost function is assigned as a combination of the states that describe the motion kinematics (position, velocity and acceleration) as will be discussed. Furthermore, a self-adapted ability is added to the developed approach where the upcoming demonstrations could recursively adapt the parameters of the learned cost function to meet the new situations.

The instantaneous state $\overrightarrow{X}_t$ of the reaching task is represented by:

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

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

$$\begin{pmatrix} \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \end{pmatrix} = \begin{pmatrix} x_t - x_{t-1} \\ y_t - y_{t-1} \\ z_t - z_{t-1} \end{pmatrix} \tag{22}$$

$$\begin{pmatrix} \ddot{x}_t \\ \ddot{y}_t \\ \ddot{z}_t \end{pmatrix} = \begin{pmatrix} \dot{x}_t - \dot{x}_{t-1} \\ \dot{y}_t - \dot{y}_{t-1} \\ \dot{z}_t - \dot{z}_{t-1} \end{pmatrix} \tag{23}$$

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

$$\overrightarrow{\dot{X}_t} = A\overrightarrow{X_t} + B\overrightarrow{u_t} \tag{24}$$

$$\overrightarrow{Y_t} = C\overrightarrow{X_t} \tag{25}$$

where $A \in \mathbb{R}^{9\times9}$, $B \in \mathbb{R}^{9\times3}$ and $C \in \mathbb{R}^{3\times9}$ are constant matrices which can be obtained easily.

4.1. Experimental setup

The Human Grasping datasets that were performed in Feix et al. (2013) are used here in order to analyze and quantify the developed PSO-based ILQR 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 a 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 started when the hand began to move and ended when the hand returned to the initial position. Two grasps were performed for each object, the first one was used to create the training data and the second one for the test data.

4.2. Preprocessing hand movements

The aim of this study is to find the cost function that describes the hand motion towards grasping an object. Hence, the used datasets have to be preprocessed before using them in the leaning step to fit the aim of our study. Firstly, the hand dorsum position $\overrightarrow{Y_t} = [x_t\ y_t\ z_t]^T$ 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 Fig. 2a. Particularly, four scenarios are involved in the experiment where a scenario is characterized by grasping a certain object (tennis ball, CD, credit card and coin) at different locations. The type of the object to be grasped is not important in this study since the location of the hand dorsum rather than the whole hand configurations is of primary interest. 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 Fig. 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 Fig. 2a. Accordingly, T_g is the translation matrix that translates the origin to the goal location $g = [g_x\ g_y\ g_z]^T$:

$$T = (T_g * R_g) \tag{26}$$

$$T_g = Trans(g_x, g_y, g_z) \tag{27}$$

$$R_g = R_{k,\theta} \tag{28}$$

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

$$R_{k,\theta} = vrrotvec(\vec{i}, \vec{g}) \tag{29}$$

which calculates the rotation matrix between two arbitrary vectors, expressed in axis/angle representation. Here, $\vec{i}$ denotes the

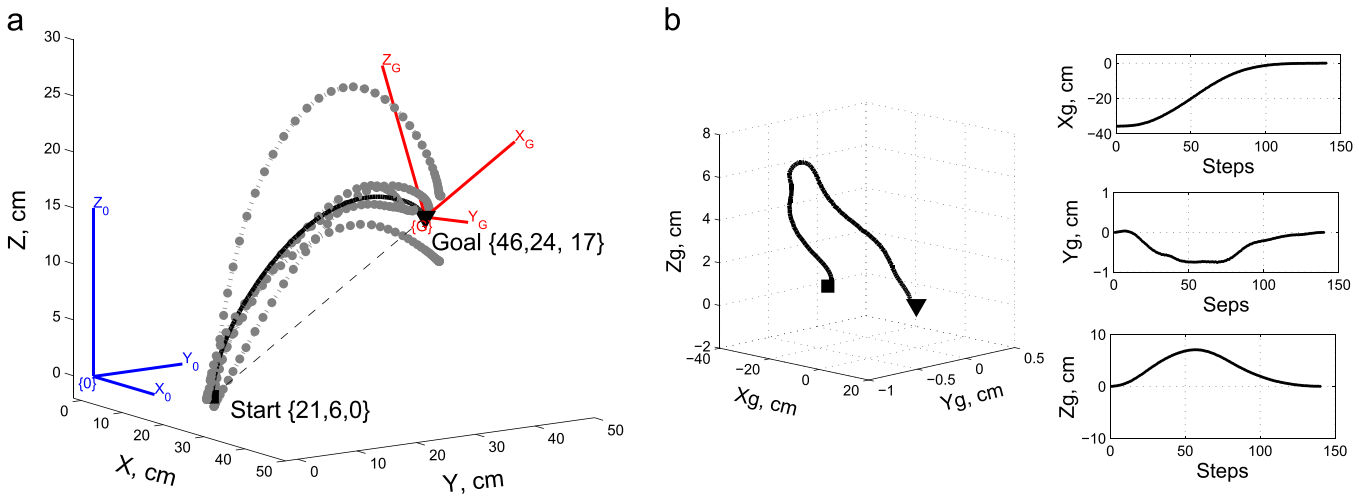

Fig. 2. (a) Averaged behavior over five subjects for the first scenario: grasping a tennis ball. (b) Transformed behavior trajectory in the goal-dependent axis system.

unit vector in the original x-axis. To summarize the preprocessing steps done over the hand trajectory datasets, the following steps are carried out sequentially:

- Extraction of the hand dorsum positions during the reaching movements.
- Computation of the averaged behavior over five subjects for each scenario.
- Projection of the average behavior to a new goal-dependent axis system to match the control context.

4.3. Reaching movement cost function

The PSO-based ILQR Algorithm 1 is applied for each of the four grasping scenarios given the preprocessed demonstrations explained above. The input parameters for the algorithm are chosen as follows: five human demonstrations per scenario (N_d=5), samples per demonstration (L=140) and a convergence threshold ($\epsilon = 2$). Each state contribution (position, velocity and acceleration) besides the control effort (jerk) is measured after obtaining the composed matrix χ that combines both the Q and R matrices as follows:

$$\chi = \begin{bmatrix} Q & \mathbf{0}_{9\times 3} \\ \mathbf{0}_{3\times 9} & R \end{bmatrix} \tag{30}$$

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 Fig. 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 (Ben-Itzhak and Karniel, 2008; Ziebart et al., 2012). They suggested that the human reaches an object with maximum motion *smoothness*, which is defined as having small high-order derivatives that is the squared acceleration in this study.

Fig. 4 shows the results of applying PSO-based ILQR 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 forward optimal problem. Trajectory in gray denotes the average measured demonstrations for each scenario used as bases for the

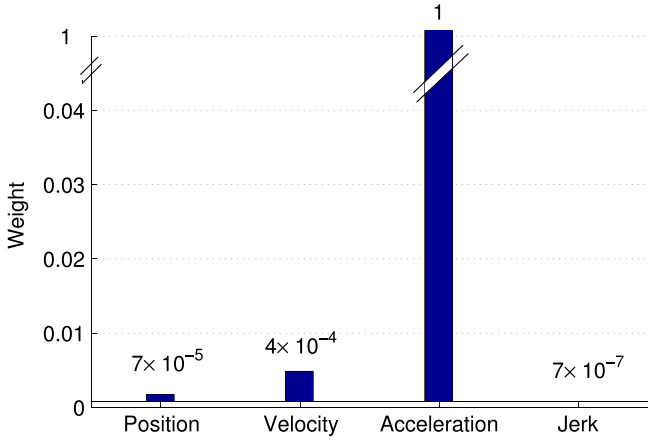

Fig. 3. Contribution of each parameters in cost function of reach-to-grasp movement (normalized by the maximum value).

computation. The discrepancies between modeled and measured behaviors in all cases are obtained in terms of the error between them in the three axes; namely e_x, e_y and e_z as shown in the sub-figures for each scenario. As depicted, error values are minimum, however, no perfect fit could be achieved since the model equations and optimization functions are always approximate ones. To quantify the generalization feature of the PSO-based ILQR approach, the obtained weighting matrices are applied with two arbitrarily test scenarios chosen from the available datasets. Namely, grasping a box and a plate scenarios. Fig. 5 shows the fit between the scenario measurements (gray line) and the estimated optimal trajectory (black line) that minimizes the cost function obtained by the PSO-based ILQR approach. As depicted, the identified cost function represents a good approximation for modeling the given test scenarios since small values of error are obtained.

Comparison of the proposed PSO-based ILQR approach with the state-of-the-art Gradient Descent (GD) approach (Priess et al., 2015) is obtained to quantify the advantages of using meta-heuristic optimization techniques. Since the proposed PSO-based ILQR and the GD-ILQR approaches have different optimization variables, the first step to validate this comparison is to enforce both techniques to start from the same initial solution. Hence, Q_0 and R_0 are chosen as the initial solution for the proposed PSO-based ILQR, while the

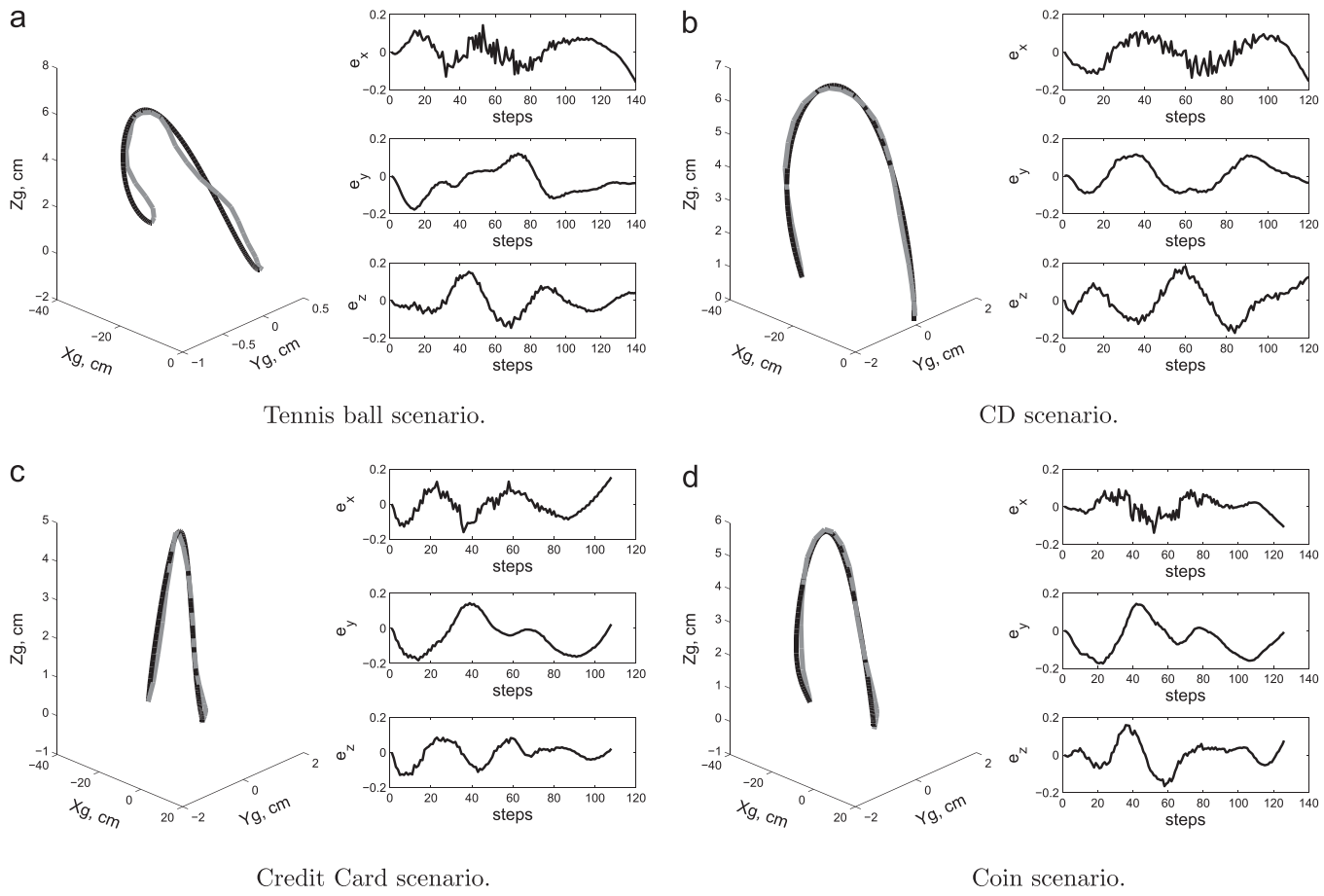

Fig. 4. Results of PSO-based ILQR performed for the four scenarios. The left plot in each sub-figure presents the 3D fit between the measurements (dashed line) with the estimated optimal trajectory (solid line) produced by the identified objective function. The other plots in the right of each sub-figure are the 2D projection in the x–y and y–z plane of the fit. (a) Tennis ball scenario. (b) CD scenario. (c) Credit Card scenario. (d) Coin scenario.

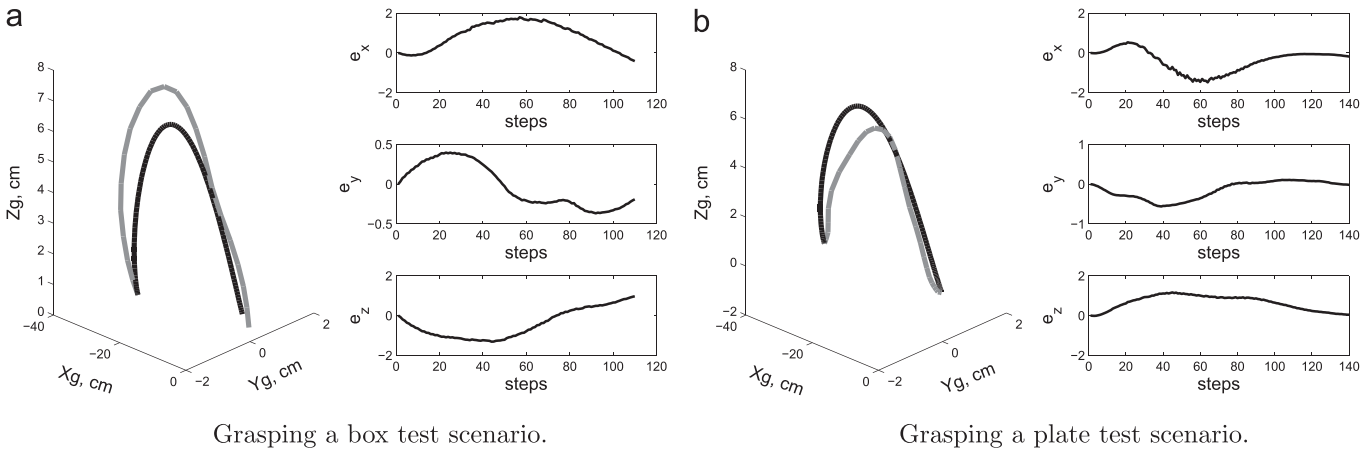

Fig. 5. Validation of the identified cost function with new test scenarios: (a) grasping a box and (b) grasping a plate. The dashed line represents the scenario measurements while the solid line.

Recatti equation, $K_0 = lqr(A, B, Q_0, R_0)$, is solved to find the initial solution K_0 for the GD-ILQR approach. Fig. 6 shows the log–log scale of the function values f_{val} given by (10) over iterations N for the mentioned GD-ILQR and the proposed PSO-based ILQR approaches. As depicted in Fig. 6, for the same initial conditions, the proposed method converges faster to small error values compared to the GD-ILQR approach. This low rate of convergence in the GD-ILQR is accounted for using the classical steepest descent optimization technique that requires a near-optimal initial solution which is not practical in our application. In addition, as stated in Section 1, GD-ILQR tries to find the cost function of a demonstrated behavior by applying two optimization steps. Namely, it searches for the closed loop gain K_e that best matches the demonstrated trajectory; then, it searches for the optimal weighing matrices Q and R that best match

the obtained gain K_e. Those double optimization steps produce a total accumulated error higher than that of the proposed method which searches for the optimal matrices that best match the measured and the estimated behaviors in one shot.

4.4. Incremental cost learning

To quantify the Evolving PSO-based ILQR 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, $\Delta\chi_s = \overline{\chi}^{(s)} - \overline{\chi}^{(s-1)}$, is measured. This indicates how the learned cost function generalizes for unseen scenarios for the behavior under study. This measure is given as follows:

$$FN^{(s)} = \| \Delta\chi_s \|_F \tag{31}$$

$$FN^{(s)} = \sqrt{\mathrm{tr}(\Delta\chi_s^T \Delta\chi_s)} \tag{32}$$

where $\mathrm{tr}(A)$ 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 differed in the starting points, number of subjects, objects to be grasped and length of demonstrations. These scenarios are selected to incrementally refine the learned composite cost function as follows:

- 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.

Fig. 7 shows the average Frobenius Norm and the average number of iterations required after applying the developed Evolving PSO-based ILQR approach with different confidence factor $\gamma_s = [0.25, 0.5, 0.75, 1]$. Fig. 7a indicates a reduction in the difference between successive learned 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 value gives small Frobenius Norm while having small number of iterations as well.

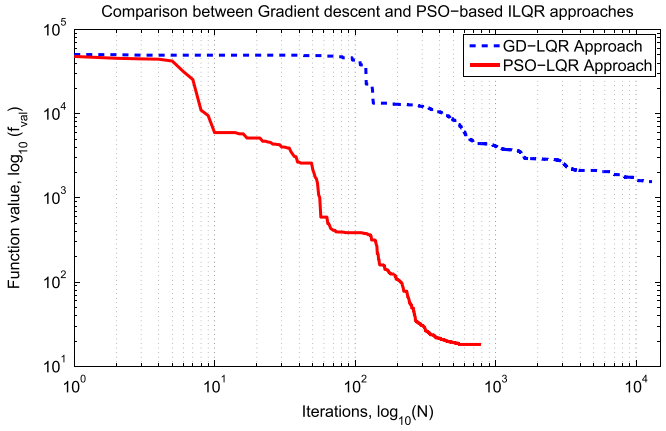

Fig. 6. Comparison of function values over iterations for both the proposed PSO-based and the Gradient Descent (GD) ILQR approaches.

5. Conclusion

A new IOC algorithm called PSO-based ILQR has been developed for learning the optimization criteria underlying a given human behavior demonstrations. The developed algorithm is based on using one of the evolutionary meta-heuristic optimization techniques called Particle Swarm Optimization. It is a derivative free and capable of finding good solutions with less computational effort. In addition, an evolving ILQR algorithm has been developed to incrementally adapt the cost function when receiving other untrained demonstrations even with different initial conditions and lengths. The developed approach has been confirmed through the application for 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, the incremental learning of the cost

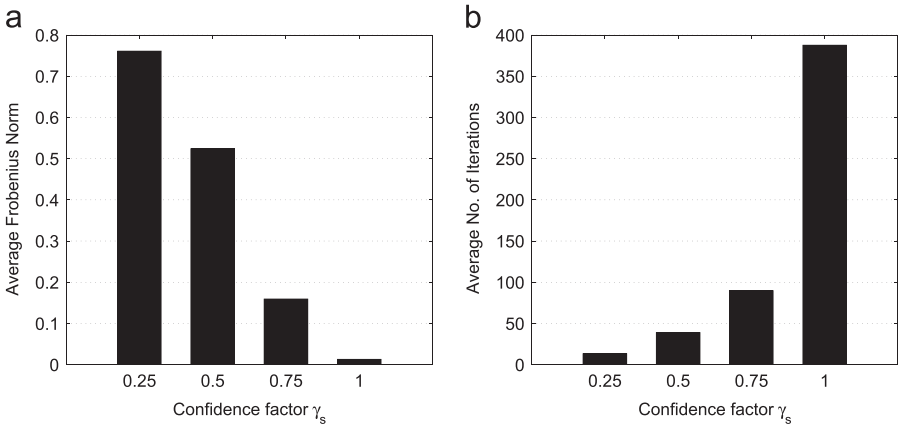

Fig. 7. Results of incremental cost learning with four different confidence factors γ_s. (a) Frobenius Norm between the composite cost matrices of successive scenarios. (b) The number of iterations required to best fit the measured scenario.

function proves reduction in the total time required for learning human behavior which provides a step toward the cost function generalization to overcome the over-fitting problem.

Acknowledgments

The first author is supported by a PhD scholarship from the Mission Department, Ministry of Higher Education (MoHE) of the Government of Egypt which is gratefully acknowledged. Also, I would like to thank the administration of Egypt–Japan University of Science and Technology (E-JUST) for their efforts to build the university as a model research university.

References

Abaid, N., Cappa, P., Palermo, E., Petrarca, M., Porfiri, M., 2012. Gait detection in children with and without hemiplegia using single-axis wearable gyroscopes. PloS One 8 (9), e73152.

Abbeel, P., Ng, A.Y., 2004. Apprenticeship learning via inverse reinforcement learning. In: Proceedings of the Twenty-First International Conference on Machine Learning. ACM, New York, NY, USA, p. 1.

Abbeel, P., Dolgov, D., Ng, A.Y., Thrun, S., 2008. Apprenticeship learning for motion planning with application to parking lot navigation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2008. IROS 2008. IEEE, Nice, France, pp. 1083–1090.

Ahmad, B., Murphy, J.K., Langdon, P.M., Godsill, S.J., Hardy, R., Skrypchuk, L., et al., 2015. Intent inference for hand pointing gesture-based interactions in vehicles. Cybernetics, IEEE Transactions on PP (99), 1. ISSN 2168-2267. < http://dx.doi.org/10.1109/TCYB.2015.2417053 > .

Atkeson, C.G., Schaal, S., 1997. Robot learning from demonstration. In: ICML, vol. 97, pp. 12–20.

Ben-Itzhak, S., Karniel, A., 2008. Minimum acceleration criterion with constraints implies bang-bang control as an underlying principle for optimal trajectories of arm reaching movements. Neural Comput. 20 (3), 779–812.

Blum, C., Roli, A., 2003. Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput. Surv. (CSUR) 35 (3), 268–308.

Boularias, A., Kober, J., Peters, J.R., 2011. Relative entropy inverse reinforcement learning. In: International Conference on Artificial Intelligence and Statistics, pp. 182–189.

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

Dragan, A.D., Srinivasa, S.S., 2013. A policy-blending formalism for shared control. Int. J. Robot. Res. 32 (7), 790–805.

El-Hussieny, H., Assal, S.F., Abouelsoud, A., Megahed, S.M., 2015. A novel intention prediction strategy for a shared control tele-manipulation system in unknown environments. In: 2015 IEEE International Conference on Mechatronics (ICM). IEEE, Nagoya, Japan, pp. 204–209.

Feix, T., Romero, J., Ek, C.H., Schmiedmayer, H., Kragic, D., 2013. A metric for comparing the anthropomorphic motion capability of artificial hands. IEEE Trans. Robot. 29 (1), 82–93, URL ⟨http://grasp.xief.net⟩.

Giraud-Carrier, C., 2000. A note on the utility of incremental learning. AI Commun. 13 (4), 215–223, URL ⟨http://dl.acm.org/citation.cfm?id=1216442.1216444⟩.

Kennedy, J., Kennedy, J.F., Eberhart, R.C., Shi, Y., 2001. Swarm Intelligence. Morgan Kaufmann, San Francisco, USA.

Khokar, K.H., Alqasemi, R., Sarkar, S., Dubey, R.V., 2013. Human motion intention based scaled teleoperation for orientation assistance in preshaping for grasping. In: 2013 IEEE International Conference on Rehabilitation Robotics (ICORR). IEEE, Seattle, WA, USA, pp. 1–6.

Kwakernaak, H., Sivan, R., 1972. Linear Optimal Control Systems, vol. 1. Wiley-Interscience, New York.

Lee, S.J., Popović, Z., 2010. Learning behavior styles with inverse reinforcement learning. In: ACM Transactions on Graphics (TOG), vol. 29. ACM, New York, NY, USA, p. 122.

Mombaur, K., Truong, A., Laumond, J.P., 2010a. From human to humanoid locomotion-an inverse optimal control approach. Autonom. Robots 28 (3), 369–383 http://dx.doi.org/10.1007/s10514-009-9170-7.

Mombaur, K., Truong, A., Laumond, J.-P., 2010b. From human to humanoid locomotion an inverse optimal control approach. Autonom. Robots 28 (3), 369–383.

Mombaur, K., Olivier, A.-H., Crétual, A., 2013. Forward and inverse optimal control of bipedal running. In: Modeling, Simulation and Optimization of Bipedal Walking, vol. 18. COSMOS. pp. 165–179.

Monfort, M., Liu, A., Ziebart, B., 2015. Intent prediction and trajectory forecasting via predictive inverse linear-quadratic regulation. In: Twenty-Ninth AAAI Conference on Artificial Intelligence.

Nakazawa, A., Nakaoka, S., Ikeuchi, K., Yokoi, K., 2002. Imitating human dance motions through motion structure analysis. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3. IEEE, Lausanne, Switzerland, pp. 2539–2544.

Newell, A., Simon, H.A., 1976. Computer science as empirical inquiry: symbols and search. Commun. ACM 19 (3), 113–126.

Ng, A.Y., Russell, S.J., et al., 2000. Algorithms for inverse reinforcement learning. In: Icml, pp. 663–670.

Priess, M.C., Conway, R., Choi, J., Popovich, J.M., Radcliffe, C., 2015. Solutions to the inverse lqr problem with application to biological systems analysis. IEEE Trans. Control Syst. Technol. 23 (2), 770–777.

Ramachandran, D., Amir, E., 2007. Bayesian inverse reinforcement learning. Urbana 51, 61801.

Soechting, J., Lacquaniti, F., 1981. Invariant characteristics of a pointing movement in man. J. Neurosci. 1 (7), 710–720.

Suleiman, W., Yoshida, E., Kanehiro, F., Laumond, J.-P., Monin, A., 2008. On human motion imitation by humanoid robot. In: IEEE International Conference on Robotics and Automation. ICRA 2008. IEEE, Pasadena, CA, USA, pp. 2697–2704.

Todorov, E., 2004. Optimality principles in sensorimotor control. Nat. Neurosci. 7 (9), 907–915.

Zhifei, S., Joo, E.M., 2012. A review of inverse reinforcement learning theory and recent advances. In: 2012 IEEE Congress on Evolutionary Computation (CEC). IEEE, Brisbane, Queensland, pp. 1–8.

Ziebart, B.D. Modeling purposeful adaptive behavior with the principle of maximum causal entropy. PhD thesis. < http://repository.cmu.edu/dissertations/17 > .

Ziebart, B., Dey, A., Bagnell, J.A., 2012. Probabilistic pointing target prediction via inverse optimal control. In: Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces. ACM, Lisbon, Portugal, pp. 1–10.