18th IEEE International Conference on Control Applications
Part of 2009 IEEE Multi-conference on Systems and Control
Saint Petersburg, Russia, July 8-10, 2009

Batch Reinforcement Learning for Semi-Active Suspension Control

Simone Tognetti, Sergio M. Savaresi, Cristiano Spelta, Marcello Restelli

Abstract— **The object of this work is the design of a control strategy for semi-active suspension. In particular this paper explores the application of Batch Reinforcement Learning (BRL) to the design problem of optimal comfort oriented semi-active suspension. BRL is an artificial intelligence technique able to provide an approximate solution of optimal control problems. The resulting control rule is a multidimensional relation which maps the measurable states of the system to the control action (reference damping). Recently a quasi optimal strategy for semi-active suspension has been designed and proposed: the Mixed SH-ADD algorithm, herein recalled for benchmarking purposes. This paper shows that an accurately tuned BRL provides a policy able to guarantee the overall best performances, which are paid in terms of complexity of both the training phase and the resulting control rationale.**

I. Introduction

Among the many different types of controlled suspension systems (see e.g. [3], [17], [21], [22], [23]), semi-active suspensions have received a lot of attention since they provide the best compromise between cost (energy-consumption and actuators/sensors hardware) and performance. The concept of semi-active suspensions can be applied over a wide range of application domains: road-vehicle suspensions, cabin suspensions in trucks or tractors, seat suspensions, suspensions in trains, suspensions of appliances (e.g. washing machines), architectural suspensions (buildings, bridges, etc.), bio-mechanical structures (e.g. artificial legs) etc ([1], [4], [6], [13], [14], [15], [24], [26]).

The research activity on controllable suspension is being developed along two mainstreams: the development of reliable, high-performance, and cost-effective semi-active controllable shock-absorbers (Electro-Hydraulic or Magneto-Rheological – see e.g. [1], [7], [9], [26], [27]), and the development of control strategies and algorithms which can fully exploit the potential advantages of controllable shock-absorbers. This work focuses on the control-design issue for road vehicles.

The design of control strategies for comfort oriented semi-active suspension can be recast into the framework of non-linear optimal control problem that cannot be solved analytically. The literature offers many contributions that provide an approximate solution of the non-linear problem, or alternatively the non-linearity is partially removed to exploit linear techniques (see e.g. [13],[17]-[20],[26]). It is a matter of fact that the numerical effort necessary for computing a solution of non-linear optimal control is often non tractable for on-line optimization and implementation on real systems.

The Batch Reinforcement Learning (BRL) is a special technique developed in the artificial intelligent research field, and provides numerical algorithms able to approximate the solution of an associated optimal control problem (see [12] and [24] for details). Such a technique is characterized by the following features, which make it extremely appealing for real application: the numerical algorithm is data based and it presents analytical and computational efforts that can be easily tackled on-line by a common microcontroller; further the BRL provides a sophisticated map that is fed by the measurable states of the system and their derivative and gives back the control action (this has much in common with Neural Networks).

In this scenario the main goals of this work are:

- To recast the optimal control problem of comfort-oriented semi-active suspension as a BRL application.
- To compare the performances provided by the control rule obtained with the BRL with the ones given by the state-of-the art semi-active control algorithms.

II. Problem statement and Previous Works.

The dynamic model of a quarter-car system equipped with a semi-active actuator can be described with the following set of differential equations (see e.g. [27]):

$$\begin{cases} M\ddot{z}(t) = -c(t)\left(\dot{z}(t)-\dot{z}_t(t)\right) - k\left(z(t)-z_t(t)-\Delta_s\right) - Mg \\ m\ddot{z}_t(t) = +c(t)\left(\dot{z}(t)-\dot{z}_t(t)\right) + k\left(z(t)-z_t(t)-\Delta_s\right) + \\ \quad -k_t\left(z_t(t)-z_r(t)-\Delta_t\right) - mg \qquad \left[z_t(t)-z_r(t)<\Delta_t\right] \\ \dot{c}(t) = -\beta c(t) + \beta c_{in}(t) \qquad c_{\min} \le c_{in}(t) \le c_{\max} \end{cases} \tag{1}$$

where the symbols in (1) are as follows (see also Fig.1). $z(t), z_t(t), z_r(t)$ are the vertical positions of the body, the unsprung mass, and the road profile, respectively. M is the quarter-car body mass; m is the unsprung mass (tire, wheel,

S. Tognetti, S.M. Savaresi and M. Restelli are with Politecnico di Milano, Dipartimento di Elettronica e Informazione, P.zza L. da Vinci 32, 20133 Milano, ITALY; (e-mail: savaresi@elet.polimi.it, tognetti@elet.polimi.it; restelli@elet.polimi.it).

C.Spelta is with the Dipartimento di Ingegneria dell'Informazione e Metodi Matematici, Università degli Studi di Bergamo, viale Marconi 5, Dalmine BG) ITALY (e-mail: cristiano.spelta@unibg.it).

978-1-4244-4602-5/09/$25.00 ©2009 IEEE

brake caliper, suspension links, etc.). k and k_t are the stiffness of the suspension spring and of the tire, respectively; Δ_s and Δ_t are the length of the unloaded suspension spring and tire, respectively. $c(t)$ and $c_{in}(t)$ are the actual and the requested damping coefficients of the shock-absorber, respectively. The damping-coefficient variation is ruled by a 1st-order dynamic, where β stands for the bandwidth; consequently the actual damping coefficient remains in that interval $c_{min} \leq c(t) \leq c_{max}$, where c_{min} and c_{max} are design parameters of the semi-active shock-absorber. This limitation is the so-called "passivity-constraint" of a semi-active suspension. For the above quarter-car model, the following set of parameters are used (unless otherwise stated): $M = 400Kg$, $m = 50Kg$, $k = 20KN/m$, $k_t = 250KN/m$, $\bar{c} = 1500Ns/m$, $c_{\min} = 300Ns/m$, $c_{\max} = 4000Ns/m$, $\beta = 30\pi$.

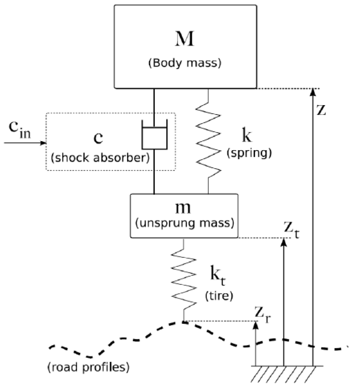

Fig.1. Quarter-car diagram.

Notice that (1) is non-linear since the damping coefficient $c(t)$ is a state variable; in the case of a passive suspension with a constant damping coefficient $\bar{c}$, (1) is reduced to a 4th-order linear system simply setting $\beta \to \infty$ and $c_{in}(t) = \bar{c}$.

The general high-level structure of control architecture for a semi-active suspension device is the following. The *control variable* is the requested damping coefficient $c_{in}(t)$. The *measured output signals* are two: the vertical acceleration $\ddot{z}(t)$, and the suspension displacement $z(t) - z_t(t)$. The *disturbance* is the road profile $z_r(t)$; it is assumed to be a non-measurable and unpredictable signal (no road preview by sonar, laser, or video-camera is available). The goal of a *comfort-oriented* semi-active control system is to manage the damping of the shock-absorber to filter the road disturbance towards the body dynamics. Thus the following cost function is introduced:

$$J = \int_0^t \left(\ddot{z}(t)\right)^2 dt \qquad (2)$$

Notice that index (2) represents the H_2 norm of the desired output. With the assumption of z_r to be a white noise this Index (2) can be viewed as the H_2 norm of the closed loop non-linear system from the road disturbance to the desired output.

It has been shown in [22] that the optimal control strategy is necessarily a rationale that switches from the minimum to the maximum damping of the shock absorber (two-state algorithms). Herein some state-of-art switching control strategies are briefly recalled.

Skyhook (SH) control.

The two-state approximation of the Skyhook control algorithm requires a two-state damper; the control law is given by ([13]):

$$\begin{cases} c_{in}(t) = c_{\max} & if\ \dot{z}(\dot{z} - \dot{z}_t) \geq 0 \\ c_{in}(t) = c_{\min} & if\ \dot{z}(\dot{z} - \dot{z}_t) < 0 \end{cases}$$

SH control is the semi-active heuristic approximation of the ideal concept of Skyhook damping (see e.g. [27]); it is the most widely used control strategy in semi-active suspension systems.

Acceleration Driven Damping (ADD) control

The implementation of ADD control requires a two-state damper; the control law is given by

$$\begin{cases} c_{in}(t) = c_{\max} & if\ \ddot{z}(\dot{z} - \dot{z}_t) \geq 0 \\ c_{in}(t) = c_{\min} & if\ \ddot{z}(\dot{z} - \dot{z}_t) < 0 \end{cases}$$

ADD control has been developed in [22], by using optimal-control theory. Interestingly enough, the SH and the ADD algorithms have a very simple (and similar) structure.

Mixed Algorithms.

Recently an almost optimal control strategy has been developed: the so called *Mixed SH-ADD* ([19]). Similarly to SH, also this strategy requires a two-state damper:

$$\begin{cases} c_{in}(t) = c_{\max} & if\ \left[(\ddot{z}^2 - \alpha^2\dot{z}^2) \leq 0 \ \wedge \ \dot{z}(\dot{z} - \dot{z}_t) > 0\right] \vee \\ & \vee \left[(\ddot{z}^2 - \alpha^2\dot{z}^2) > 0 \ \wedge \ \ddot{z}(\dot{z} - \dot{z}_t) > 0\right] \\ c_{in}(t) = c_{\min} & if\ \left[(\ddot{z}^2 - \alpha^2\dot{z}^2) \leq 0 \ \wedge \ \dot{z}(\dot{z} - \dot{z}_t) \leq 0\right] \vee \\ & \vee \left[(\ddot{z}^2 - \alpha^2\dot{z}^2) > 0 \ \wedge \ \ddot{z}(\dot{z} - \dot{z}_t) \leq 0\right] \end{cases}$$

Notice that the key idea is condensed in $\ddot{z}^2 - \alpha^2\dot{z}^2$; accordingly to its sign, it selects an appropriate sub-strategy. This quantity can be seen as a frequency selector and α (the unique tuning knob) represents the desired cross-over frequency between two suboptimal strategies, namely SH and ADD. Readers are referred to [19] for a formal explanation of frequency range selector $\ddot{z}^2 - \alpha^2\dot{z}^2$. A single sensor implementation of this strategy has been recently developed (the so called *1-Sensor-Mix*, [20]).

These three algorithms above have been already compared in the time and frequency domains ([19]). A picture of the comparison between SH, ADD, and Mixed SH-ADD control strategies is presented in Fig.2, where the approximated frequency responses from the road profile to the body acceleration are shown. Notice the concept of

Frequency Response (FR) cannot be adopted by System (1) due to its non-linear nature. However approximations of FR can be exploited for analysis, such as describing function or variance gain (see. e.g. [17] and [9]). In this work the Variance Gain tool has been adopted for evaluation purposes.

In Fig.2 it is also introduced a lower bound (black dot line), representing the best possible performance achievable with a semi-active system with a full knowledge of the road disturbance, but non-feasible (see [19] for further details). By inspecting Fig. 2 some conclusions can be drawn:

- The so called passive trade-off appears evident. The system with low damping shows a good filtering at high frequencies, whereas a bad resonance appears at low frequency (body resonance). The body resonance is well damped by a hard shock absorber, but unfortunately this is paid in terms of extremely bad filtering at high frequency.
- At low frequencies the SH provides the best possible filtering performances;
- At high-mid frequencies the ADD provides the best possible filtering performances;
- Compared to passive suspension, a controlled system is able to remove the passive trade off. In other words it is possible over a band of interest to outperform a passive system without deteriorating the performances elsewhere.
- the Mix-SH-ADD inherits the *best behavior* of both SH and ADD in their frequencies range of optimality. In these terms this algorithm seems to provide the almost optimal performances for comfort-oriented semi-active suspensions.

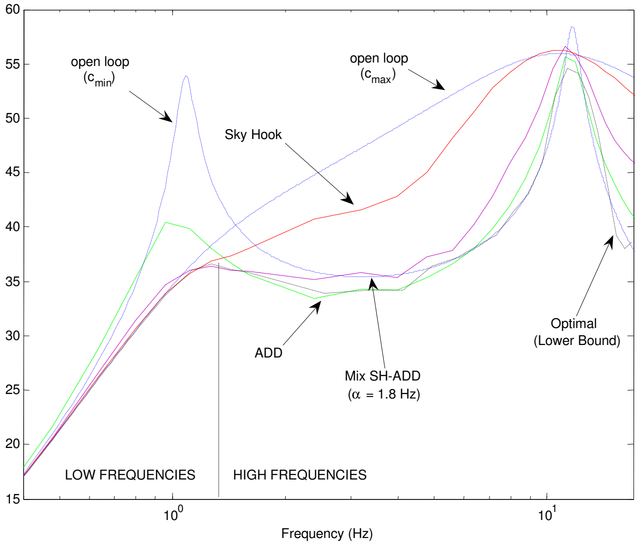

Fig.2. Comparison of the filtering performance of SH, ADD, Mix SH-ADD, and optimal lower bound.

III. Batch Reinforcement Learning and its application to semi-active suspensions

Research in Reinforcement Learning (RL) aims at designing training algorithms which make autonomous agents (controllers) able to *learn* how to behave (definition of a control policy) in a particular *environment* (controlled system) starting from the *interaction* between the agent's *action* (control input) and the enviroment response (measurable outputs). See e.g [24] and [12] for a broad overview.

The interaction between the agent and the environment is modeled as a discrete-time Markov Decision Process (MDP). An MDP is a tuple $< S, A, P, R, \gamma >$, where S is the state space, A is the action space, $P: S \times A \to \Pi(S)$ is the transition model that assigns to each state-action pair a probability distribution over S, $R: S \times A \to \Pi(R)$ is the reward function, or cost function, that assigns to each state-action pair a probability distribution over R, $\gamma \in [0,1)$ is the discount factor. At each step, the agent chooses an action according to its current *policy* $\pi: S \to \Pi(A)$, which maps each state to a probability distribution over actions. The goal of an RL agent is to maximize the expected sum of discounted rewards, that is to learn an optimal policy π^* that leads to the maximization of the action-value function, or cost-to-go from each state.

The optimal action-value function $Q^*(s(t), a(t)), s(t) \in S, a(t) \in A$ is defined by the Bellman equation:

$$Q^*(s(t),a(t))= \sum_{s(t+\Delta T)\in S} P(s(t+\Delta T)|s(t),a(t)) \left[R^*(s(t),a(t))+\gamma \max_{a(t+\Delta T)\in A} Q^*(s(t+\Delta T),a(t+\Delta T))\right] \qquad (3)$$

where $R^*(s,a) = E[R(s,a)]$. From the Control Theory perspective this equation represents the optimal cost-to-go, indeed it represents the discrete-time version of the Hamilton-Jacobi-Bellman equation.

One of the main advantages of RL algorithms is their ability to work without prior knowledge of system dynamic. This means that a full knowledge of the transition model P is not necessary for training. The algorithm, indeed, does not solve analitically the Bellman equation associated, but estimates the solution by using samples defined as $< s(t), a(t), s(t+1), r(t) >$ gathered from the interaction with the system. Each sample is constituted by the agent's action $a(t)$, a measure of the system state $s(t)$ before the action, a measure of the system state after the action $s(t+1)$, and the immediate reward $r(t)$.

In order to manage the huge amount of experience (samples) needed to solve a real-world task, batch approaches have been proposed in [2], [5] and [18]. The main idea is to distinguish between the exploration strategy that collects samples (sampling phase), and the offline learning algorithm that, on the basis of the collected data, computes the approximation of the action-value function (learning phase). Finally the latter phase provides the closed loop control rationale.

Let us consider a system having a discrete-time dynamics. If the transition model or the reward function are unknown, we cannot use dynamic programming to solve the control problem. However, we suppose to perform a collection phase to obtain data from one or more system trajectories generated starting from an initial state, following a given policy, namely:

$$F=\{<s(t)^i,a(t)^i,s(t+1)^i,r(t)^i>,i=1..K\} \quad (4)$$

The learning phase depends on the chosen RL algorithm. We focused Fitted techniques ([2], and references therein) that reformulate the estimation of the value function as a sequence of regression problems. Given the set of samples *F*, Fitted Q-Iteration (FQI) estimates the optimal action-value function by iteratively extending the optimization horizon ($Q_N - function$). First Q_0 is initialized with $Q_0(s,a) = 0$, then the algorithm iterates over the full sample set *F*. Let us consider the i-th sample $< s(t)^i, a(t)^i s(t+1)^i, r(t)^i >$ given approximation of the *Q*-function at time N (Q_N), the estimation of Q_{N+1} is performed by using classical *Q*-learning update rule ([28]):

$$Q_{N+1}(s(t)^i,a(t)^i) = (1-\alpha)Q_N(s(t)^i,a(t)^i) + \alpha(r(t)^i + \gamma \max_{a' \in \mathsf{A}} Q_N(s(t+1)^i,a')) \quad (5)$$

Since $Q_0 = 0$, the first iteration estimates only the mean reward for a state-action pair. The maximization is performed going through all possible actions since we are supposing to have a discrete action space.

Actually, for each sample a new one is generated replacing single step reward by $Q_1 =< s(t)^i, a(t)^i s(t+1)^i$, $Q_1(s(t)^i . a(t)^i >$. This intermediate sample-set defines a regression problem from $s(t)^i . a(t)^i$ to $Q_1(s(t)^i . a(t)^i$ that enables the estimation of Q_1. Thereafter, at each iteration N, corresponding to a N-step horizon, a new regression problem is defined, in which the training samples are computed exploiting the approximation of the action-value function at the previous iteration.

A. Function approximation

Tree-based regression methods are used to approximate a function by using a top-down approach. Each method produces one or more trees (*ensemble*) that are composed by a set of decision nodes used to partition the input space. The tree determines a constant prediction in each region of the partition by averaging the output values of the elements of the training set $TS = \{(i^1, o^1), \dots, (i^{\#TS}, o^{\#TS})\}$ which belong to this region.

In the particular case of the *Q*-function approximation, *TS* is derived from the sample set (5). The input space is defined by $i^l = < s(t)^l, a(t)^l >$ and the output o^l is the *Q*-function value associated to the value of i^l.

In this work, we used extremely-randomized tree ensemble (Extra-Tree Ensemble [8]) that is composed by a forest of *M* trees where each tree is constructed by randomly choosing *K* cut-points i_j, representing the *j*-th component of the action-state space, and the correspondingly binary split $[i_j < t]$, representing the cut-direction. The construction proceeds by computing a score for each test, choosing the one that maximizes the score. The algorithm stops splitting a node when the number of elements in this node is lower than a parameter n_{min}.

B. Problem definition

When RL is used to in control application, the following elements have to be defined:

- The measurable signals that define the state space of the quarter car system.
- the action variable representing the control variable.
- the reward function representing the control objective.
- the learning algorithm, with its own parameters, used to estimate numerically the solution of Bellman equation.

In this scenario we consider a 3-dimensional state space $\mathsf{S}_3 \equiv < \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >$, which correspond to the measurable states of the suspension system, namely the body vertical acceleration $\ddot{z}$, the body velocity $\dot{z}$ (obtained by integrating the corresponding acceleration), and the suspension stroke velocity $\dot{z} - \dot{z}_t$.

Since a comfort oriented semi-active optimal policy is an ON-OFF switch policy (see e.g. [19]). The action space can be defined as a two-state space, namely:

$$A =< c_{in}(t) > \ \mid \ c_{in}(t) \in \{c_{min}, c_{max}\} \quad (7)$$

The minimization of the squared vertical accelerations (2) can be defined as the maximization of following reward:

$$\mathsf{R}_1(< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >, < c_{in}(t) >) = -\ddot{z}^2(t + \Delta T) \quad (8)$$

where $\ddot{z}(t+\Delta T)$ is the accleration obtained by integrating system (1) over ΔT from the initial condition given by the states values at time *t*. During ΔT the action $c_{in}(t)$ and the road profile $z_r(t)$ are supposed to be fixed.

In practice the ideal goal of a semi-active suspension system is to negate the body vertical movements around its steady state conditions, with respect to any road disturbance. The formaql represantation of this goal by index (2) makes the associated control problem more tractable, however in the RL domain the following rewards can be also taken into account:

$$\mathsf{R}_2(< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >, < c_{in}(t) >) = -\dot{z}^2(t + \Delta T) \quad (9)$$

$$\mathsf{R}_3(< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >, < c_{in}(t) >) = -\left(z(t+\Delta T) - \bar{z}\right) \quad (10)$$

Where the used notation and meanings have been prevoisly introduced. Rewards (9) and (10) aim to minimize the squared variation of the body vertical velocity and the squared variation of the body vertical position, respectively.

C. Algorithm parameters

By using FQI, we need to define the sampling phase, indeed, we have to choose how samples are collected, how many samples we would like to use, and the fitted horizon. The samples are generated by feeding System (1) with a road disturbance z_r designed as a integrated band limited white noise. This kind of signal is a realistic approximation of a road profile and it is able to excite all the system dynamics ([11]). Generally speaking Fitted techniques assume no constraints on the characteristics of the distrubance.

System is then initialized with the steady state conditions that can be easily computed form equations (1). Simulation is performed by using a fixed step-size solver that integrates numerically. The simulation test lasts 300seconds, which seems to be the best trade-off between the performance of training and computational costs.

The optimization horizon (fitted length) suffers of a clear trade-off: the longer the horizon the more accurate the trianing results are. Unfortunately this is paid in terms of computational complexity. The deep analysis on the best compromise is here omitted for the sake of conciseness

The regressor parameters need also to be chosen. The number M of trees depends mainly on the complexity of the problem and affects the time cost; we used 20-trees ensemble. Experiments in [8] have shown that a good default value for the parameter K (the number or cut points to be randomly chosen) is actually the dimension of the input space: $K = |S| + |A| = 4$. The number of samples into a leaf depends mainly on the generalization we would like to provide and we choose $n_{min} = 100$.

IV. Numerical Results

The BRL is trained for the three proposed cost functions $\mathrm{R}_1, \mathrm{R}_2, \mathrm{R}_3$ and for several optimization horizons. The performance are evaluated by using the standard cost function based on the minimization of the squared variations of body vertical accelerations. Interestengly enough the best result is achieved if the reward based on the body vertical velocity variation R_2 is used. This is mainly due the numerical approximations of the BRL, however this aspect is currently under deep investigation.

The results of the BRL optimization is a multidymensional control map that associates every measured sample set $< \ddot{z}(t), \dot{z}(t), \dot{z}(t) - \dot{z}_t(t) >$ to a control action $c_{in}(t)$. Notice that if a sample set is not present in the map entries, a nearest -neighbour approach is used for delivering an appropriate control action. A graphical represantation of this map is depicted in Fig. 3, where it is compared to the one obtained by controlling the semi-active system with the Mixed SH-ADD rule (quasi optimal algorithm).

By inspecting Fig. 3 the following considerations can be done:

- The BRL rule mapping is very similar to the one associated to the Mixed SH-ADD algorithm. This was expected since the Mixed-SH-ADD is an optimal approximation as well as the BRL provide an approximation of optimal control map for semi-active suspension.
- BRL map tends to prefer a high damped suspension,. The main differences betweeen BRL map and Mixed SH-ADD can be highlighted around the orgin of the axis. However notice that in such a situation any selected damping cannot influences the body dynamics.

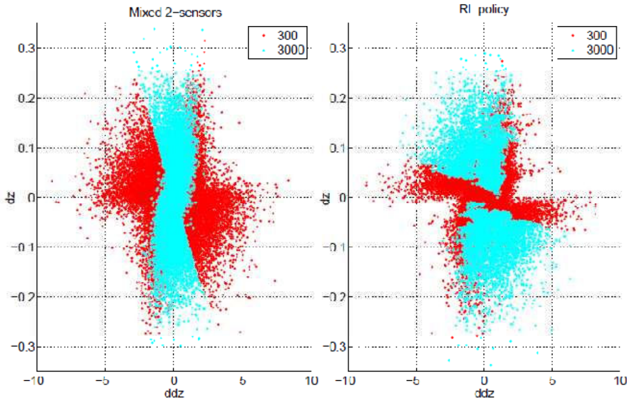

Fig. 3a. Graphical represantation of Mixed SH-ADD algorithm (left) and BRL policy (right) over the $\ddot{z}, \dot{z}$ plane

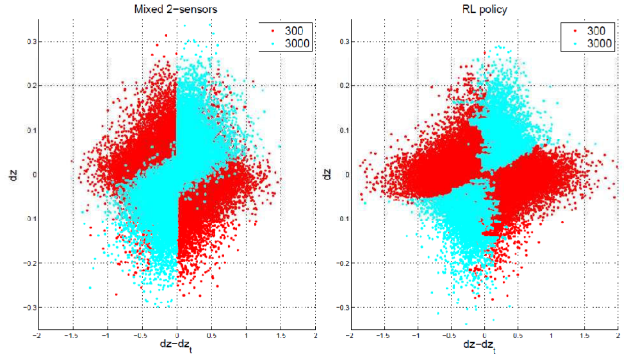

Fig. 3b. Graphical represantation of Mixed SH-ADD algorithm (left) and BRL policy (right) over the $(\dot{z} - \dot{z}_t), \dot{z}$ plane

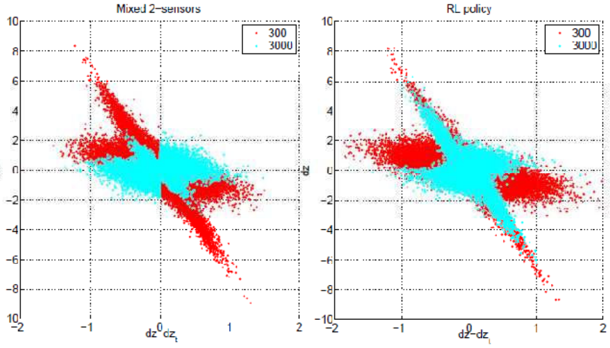

Fig. 3c. Graphical represantation of Mixed SH-ADD algorithm (left) and BRL policy (right) over the $(\dot{z} - \dot{z}_t), \ddot{z}$ plane

The performances of the semi-active suspension system fed with a random signal $z_r(t)$ and ruled by the BRL have been evaluated in time and frequency domain.

The frequency domain analysis in reported in Fig. 4, which depicts the approximate frequency response obtained as the ratio between the power spectrum of the output $\ddot{z}(t)$ and the input $z_r(t)$.

The time domain results are condensed in Fig. 5 where the performance index (2) is reported for different control strategy and compared to the extreme passive configuration.

By inspecting Fig.4 the following conclusion can be done:

- The BRL policy outperforms the Mixed SH-ADD at low frequency. This is paid in terms of filtering at high frequencies where Mixed SH-ADD shows a better behavior.

- Overall the BRL provides the best results in terms of minimization of integral of squared vertical body accelerations.

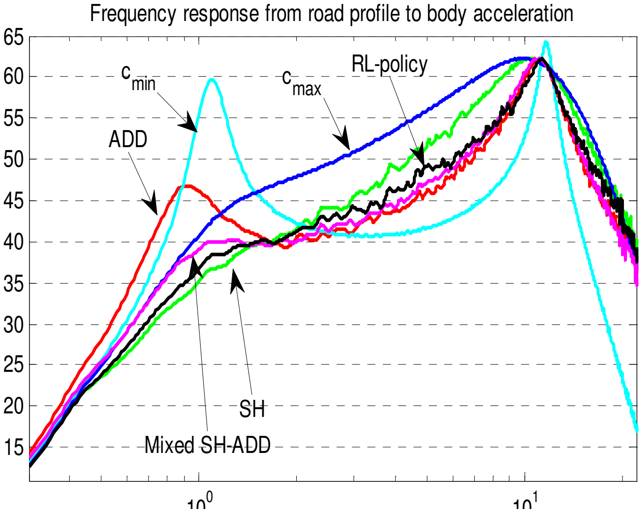

Fig. 4. Evaluation in the frequency domain of open loop configurations (low and high damping), Skyhook, Mixed SH-ADD and RL policy.

V. Conclusions

This paper has presented a new method for solving optimal control problem related with an application to semi-active suspension: the batch reinforcement learning (BRL), developed in the research area of artificial intelligence. The resulting control rule is a multidimensional relation which maps the measurable inputs to the optimal control action. It has been shown that an accurately tuned BRL provides a policy able to guarantee the overall best performances (compared to the state-of-art existing semi-active control strategies), which are paid in terms of complexity of both the training phase and the resulting control rationale. An experimental activity is planned to confirm the present results obtained numerically.

References

[1] Ahmadian M., B.A. Reichert, X. Song (2001). System Nonlinearities Induced by Skyhook Dampers. Shock and Vibration, Vol..8, No.2, pp.95-104.

[2] A. Antos, R. Munos, and C. Szepesvari, “Fitted q-iteration in continuous action-space mdps,” in Advances in Neural Information Processing Sys-tems 20, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. Cambridge, MA: MIT Press, 2008, pp. 9–16.

[3] Campi M.C., Lecchini A., Savaresi S.M. (2003). An application of the Virtual Reference Feedback Tuning (VRFT) method to a benchmark active suspension system. European Journal of Control, Vol..9, pp.66-76.

[4] Caponetto R., O. Diamante, G. Fargione, A. Risitano, D. Tringali (2003). A soft computing approach to Fuzzy Sky-Hook control of semi-active suspension. IEEE Transactions on Control System Technology, Vol.11, No. 6, pp.786-798.

[5] D. Ernst, P. Geurts, L. Wehenkel, and L. Littman, “Tree-based batch mode reinforcement learning,” Journal of Machine Learning Research, vol. 6, pp. 503–556, 2005.

[6] Giuia A., C. Seatzu, G. Usai (1999). Semiactive suspension design with an optimal gain switching target. Vehicle System Dynamics, Vol.31, No.4, pp. 213-232.

[7] Goodall R.M., W. Kortüm (2002). Mechatronic developments for railway vehicles of the future. Control Engineering Practice, Vol.10, pp.887-898.

[8] P. Geurts, D. Ernst, and L. Wehenkel, “Extremely randomized trees,” Machine Learning, vol. 63, no. 1, pp. 3–42, 2006.

[9] N. Giorgetti, A. Bemporad, H. E. Tseng , D. Hrovat (2005). Hybrid Model Predictive Control Application Towards Optimal Semi-Active Suspension. Preceedings of IEEE ISIE 2005, Dubrovnik, Croatia.

[10] Guardabassi G.O., S.M. Savaresi (2001). Approximate Linearization via Feedback - an Overview. Survey paper on Automatica, Vol..27, pp.1-15.

[11] Hrovat, D. (1997). Survey of Advanced Suspension Developments and Related Optimal Control Applications. Automatica, Vol..33, n.10, pp. 1781-1817.

[12] Kaelbing L.P., M.L. Littman, A.W. Moore (1996). Reinforcement Learning: a Survey. Journal of artificial Intelligence Reasearch, vol. 4, pp 237-285, 1996.

[13] Karnopp, D. C., M.J. Cosby (1974). System for Controlling the Transmission of Energy Between Spaced Members. U.S. Patent 3,807,678.

[14] Kawabe T., O. Isobe., Y. Watanabe, S. Hanba, Y. Miyasato (1998). New semi-active suspension controller design using quasi-linearization and frequency shaping. Control Engineering Practice, Vol..6, No.10, pp. 1183-1191.

[15] Kiencke U., Nielsen L. (2000). Automotive Control Systems for Engine, Driveline, and Vehicle. Springer Verlag.

[16] Liao, W.H., D.H. Wang (2003). Semiactive Vibration Control of Train Suspension Systems via Magnetorheological Dampers. Journal of Intelligent Material Systems and Structures, Vol.14, No. 3, pp.161-172.

[17] Sammier D., Sename O., Dugard L. (2003). Skyhook and H_∞ control of semi-active suspensions: some practical aspects. Vehicle System Dynamics, Vol.39, n.4, pp. 279-308..

[18] M. Riedmiller, “Neural fitted q iteration - first experiences with a data efficient neural reinforcement learning method,” in ECML, 2005, pp. 317– 328

[19] Savaresi S.M., C. Spelta (2007). Mixed Sky-Hook and ADD: Approaching the Filtering Limits of a Semi-Active Suspension. ASME transactions: Journal of Dynamic Systems, Measurement and Control, Volume 129, Issue 4, 382.

[20] Savaresi S.M., C. Spelta (2008). A Single Sensor Control Strategy for Semi-Active Suspension. To Appear.

[21] Sayers M.W. (1999). Vehicle models for RTS applications. Vehicle System Dynamics, Vol.32, pp.421-438.

[22] Savaresi S.M., E. Silani, S. Bittanti (2005). Acceleration-driven-damper (ADD): an optimal control algorithm for comfort-oriented semi-active suspensions. ASME Transactions: Journal of Dynamic Systems, Measurement and Control, vol.127, n.2, pp.218-229.

[23] Silani E., S.M. Savaresi, S. Bittanti, A. Visconti, F. Farachi (2003). The concept of performance-oriented yaw-control systems: vehicle model and analysis. SAE Transactions, Journal of Passenger Cars - Mechanical Systems. Vol.2002, ISBN No.0-7680-1290-2, pp.1808-1818.

[24] Sutton R., A. Barto. (1998). Reinforcement Learning: an introduction. MIT Press.

[25] Tseng H.E., J.K. Hedrick (1994). Semi-active control laws – Optimal and Sub-Optimal. Vehicle System Dynamics, Vol.23, pp.545-569

[26] Valasek M., W. Kortum, Z. Sika, L. Magdolen, O. Vaculin (1998). Development of semi-active road-friendly truck suspensions. Control Engineering Practice, Vol. 6, pp.735-744.

[27] Williams R.A. (1997). Automotive Active Suspensions Part 1: basic principles. IMechE, Vol. 211, Part D, pp. 415-426.

[28] C. Watkins, “Learning from delayed rewards,” Ph.D. dissertation, Cam- bridge University, Cambridge,England, 1989.