WSES/IEEE Conference on Neural, Fuzzy and Evolutionary Computation, CSCC 2001, Rethymnon, Crete, July 2001

# Non Parametric Learning of Sensori-Motor Maps. Application to the Control of Multi Joint Systems

MARTEAU PIERRE-FRANCOIS, GIBET SYLVIE, JULLIARD FREDERIC
Valoria
South Britany University
Rue Yves Mainguy, 56000 Vannes
France
{pierre-francois.marteau, sylvie.gibet, frederic.julliard}@univ-ubs.fr

*Abstract:* - At the light of control and learning theories, this paper addresses the question of controlling multi-joint system using sensory feedback. A generic Sensory-Motor Control Model (SMCM) is firstly presented that solves the inverse kinematics difficulty at a theoretical level. Computational implementations of SMCM requires the knowledge of sensory motor transforms that are directly dependent to the multi-joint structure that is to be controlled. To avoid the dependency of SMCM to the analytical knowledge of these transforms, a non parametric learning approach is developed to identify non linear mappings between sensory signals and motor commands involved in SMCM. The resulting adaptive SMCM (ASMCM) is intensively tested within the scope of hand-arm reaching movements. ASMCM shows to be very effective and robust at least for this task. Its generic properties and effectiveness allow to foresee wide area of application.

*Key-Words:* - Non Parametric Learning, Sensory-Motor Maps, Multi-joint System Control, Adaptive Control System.

## 1 Introduction

Sensory-motor controlled systems exhibit non linear mappings between sensory signals and motor commands, that are in many respects basic components involved in the control of complex multi-articulated chains. To cope with the need for plasticity (adaptation to changes), generic control performance (similar control principles for various kinds of mappings, various kinds of articulated chains or neuro-anatomical variability among individuals) and anticipation (predictive capability and optimization of movements) it is more or less accepted in the neuro-physiology community that these mappings are learned by biological organisms rather than pre-programmed. For the design of artificial system control, the biological plausibility of the control mechanisms involved is not really considered as an issue. Nevertheless, adaptive, predictive and generic capabilities of controlling components are indeed key characteristics that have been carefully addressed for a long time, in particular within the optimal control and robotics communities.

Furthermore, in the scope of complex artificial system design, analytical equations that drive the dynamics and the kinematics of the system could be difficult to extract, and the corresponding solution to the set of differential equations fastidious to estimate. Setting up control strategies for complex system control is consequently not a simple task. In this context, learning part of the control strategy from the observation of the system behavior could be an appealing and efficient approach. The aim of this paper is to present a generic learning approach for the control of sensory motor systems.

## 2 Controlling Sensory-Motor Systems

Any motor system can be characterized in a state (or phase) space where the state of the system is supposed to be completely determined at any time by a point in this space. Time evolution of the system result in the development of a trajectory in the state space. For sensory-motor systems, the state is observed through sensory signals (observable outputs), as shown in Fig. 1. Thus, the track of the evolution within the state space results in the development of a trajectory within a sensory space.

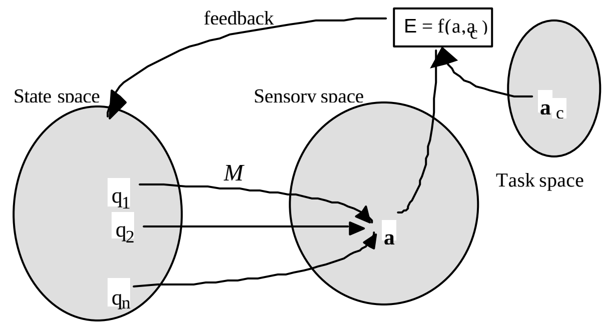


*Fig. 1 : System controlled with sensory feedback*

The mapping $M$ between sensory outputs and state characterizations is generally highly non linear and projective (the dimensionality of the state space is higher than the dimensionality of the sensory space). When the system is controlled using a sensory feedback, the task affected to the system is specified in a space homogeneous to the sensory space labeled task space in figure 1. The error signal measured between sensory outputs and task inputs is finally used as a feedback to update the state of the system. The projective transforms *M* are generally well-defined functions, with a redundancy in the articulated systems characterized by an excess of degrees of freedom, e.g. the transformation between the input and the output is characterized by a many-to-one transform. Thus, the same sensory outputs may be observed for numerous different states of the system. Consequently, forward mapping *M* is well-defined while the inversion of *M* is an ill-posed problem. Control strategies of Sensory Motor Systems (SMS) require some how either the knowledge of the forward mapping *M* or the knowledge of its pseudo inverse: $M^{-1}$. Both are closely related to the structure of the SMS itself.

# 3 The Sensori-Motor Control Model (SMCM)

Numerous solutions exist to control SMS (see references [1], ..,[5] for some approaches that exploit learning mechanisms. We present here a general model that has been proposed to control various kind of articulated chains, in particular hand-arm systems [6]. For this model, the update of the system state $q$ is computed on the basis of the error $E$ between the current sensory outputs $a$ and the task specification $a_c$, according to a gradient descent strategy.

$$\frac{\partial q}{\partial t} = -g(E(a,a_c,t)).\frac{\nabla(E(a,a_c,t))}{\left\|\nabla(E(a,a_c,t))\right\|}$$

$$= -g(E(a,a_c,t)).\frac{(\frac{\partial M}{\partial q}).(M(q)-a_c)}{\left\|(\frac{\partial M}{\partial q}).(M(q)-a_c)\right\|} \quad (1)$$

$(\frac{\partial M}{\partial q})$ is the Jacobian matrix of the operator $M$, $g$ is a gain function and $\nabla$ the gradient operator.

In order to ensure the stability of the system and to generate damped behaviors, a nonlinear function g and a second order filter have been introduced. The nonlinear function has a sigmoïd shape: the gain of this function increases significantly when the error between the observable position and the reference target position goes towards zero. Stability and asymptotic properties of such a model have already been studied in [6].

For SMCM, the *M* transform enters in the computation of the gradient of the error signal $\|a-a_c\|$. To implement this model, all coefficients of the jacobian matrix $(\frac{\partial M}{\partial q})$ should be known for all values of the state vector $q$. These coefficients depend directly on the structure of the articulated chain to be controlled. For any articulated chain, a specific jacobian matrix should be calculated.

# 4 Learning sensory-motor mappings involved in SMCM

## 4.1 Non Parametric Learning v.s. Parametric Learning

Two distinct and competing approaches are available when facing the problem of learning non linear transforms (NLT) and in particular non linear mappings involved in multi-joint control systems: parametric learning (PL) and non parametric learning (NPL) (Cf. [7] for a pioneer and detailed synthesis on PL and NPL, and [8] for a more recent review of PL and NPL models with biological relevance arguments regarding internal sensory-motor maps). The fundamental difference between PL and NPL is that PL addresses the learning essentially globally while NPL addresses it much more locally. In other words, PL methods try to learn non linear transforms over their whole domain of validity. This means that if a change in the environment occurs locally, it will potentially affect the learning process every where in the definition domain of the transform. Conversely, NPL learns the properties of the transform in the neighborhood of each point of interest within the definition domain of the transform. Thus, a local update in the learning process does not affect the rest of the learned definition domain. Neuromimetic networks are an instance of the PL class with synaptic weights as parameters, while near neighbors derived methods are instances of the NPL class.

Biological relevance can be found for the two kinds of approaches [9]. Nevertheless, local characteristic of NPL is undoubtedly a great advantage when addressing incremental learning in variable environments, since the local modification resulting from any change does not affect the overall structure of the NLT already learned.

### 4.2 Learning SMCM maps using NPL approach

To approach the normalized gradient of the error, the following map $f$ is defined:

$$d\hat{q} = f(q, da) \qquad (2)$$

Where $da$ is the 3D directional vector toward the target position specified in the 3D Cartesian space, $q$ the vector of the joint variables that characterize the articulated chain. $d\hat{q}$ is the estimated normalized modification within the state space that will minimize the distance between the articulated chain end point and the position of the target $a_c$. The calculation of the map $f$ is approximated through a variable gaussian kernel density estimator as explained below:

Given a set $N$ of learning samples, $\{(q_i, dq_i, da_i)\}_{i=1...N}$ , the state update $d\hat{q}$ that minimizes the error signal calculated from a current state $q$ and a 3D normalized directional vector $da$ is estimated as follows:

$$d\hat{q} = \frac{\sum_{i=1}^{N} K(x_i, x) dq_i}{C}, \qquad (3)$$

where $x=[q, da]$ , $x_i=[q_i, da_i]$, C is a normalizing factor, and $K$ a variable gaussian kernel:

$$K(x_i, x) \approx \exp\left( -\frac{(x - x_i) W (x - x_i)^T}{s} \right) \qquad (4)$$

$W$ is a weighting diagonal matrix used to balanced the importance of sensory inputs,($da$) vs. motor data ($q$), $s$ is a variable that depends on the local densities in the state space and in the sensory space: if the density is low, $s$ is increased and conversely, if the density is high, $s$ is lowered.

### 4.3 Naive learning algorithm

Initialisation: select a small value $\varepsilon$, an integer value $N$ and set $i$ to $0$ ($e$ can be a function of $N$).

1) Select randomly a state vector $q$, position the multi-joint system according to $q$, and observe the corresponding sensory outputs $a$.
2) Select a small normalized change $dq$, position the multi-joint system according to $(q + dq)$, and observe the change in sensory outputs $da$.
3) Calculate $d\hat{q}$ using $x=W.[q, da]^T$ according to equations (3) and (4).
4) If $||d\hat{q} - dq|| > e$, save the association $([q, da], dq)$ as a new learning sample ($\xi_i$ , $dq_i$) and increment $i$.
5) If $i<N$, loop in 1), stop otherwise

### 4.4 Implementation issues

In estimating the expectation of the state update ($dq$), the computations of the distances in the d-dimensional (dimension of the state space + 3) space of $x$ vectors are required. In the sum involving gaussian kernels (eq. (3) and (4)), only the $\xi_i$ vectors belonging to the neighborhood of $\xi$ are retained. A kd-tree [10] for identifying these neighborhoods in logarithmic time with N.

## 5 Experiments

### 5.1 Controlling a geometric arm model

To exemplify the flexibility and the generic characteristics of ASMCM we present a simple experiment carried out on a simulated arm system submitted to a reaching task. The arm is composed of three joints with 7 degrees of freedom as shown in figure 2. It is controlled by a ASMCM where the gradient descent strategy has been estimated using the learning of the non linear mapping $f$ characterized in the previous section.

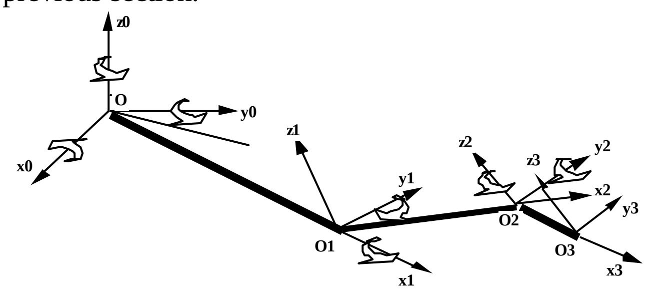


*Fig. 2 : Geometric arm model with three joints and seven degrees of freedom. The length of the joints are respectively .4, .4 and .2, in simulation units*

To evaluate the learning capability of the above presented NPL technique, ASMCM has been tested as follows:

For increasing values of the number of learning samples $N$, five learning runs are carried out. For each of these runs, two hundred and fifty 3D spatial target positions and initial conditions have been selected randomly. For these *250* conditions, the error rate (number of cases where the arm is not able to reach the target), the average of the time to target (number of iterations of the sensory motor loop) and the average of the residual distance to target when errors occur are calculated.

The experimental settings for this test are the following:

- A target is considered to be reached when the residual distance between the arm end-point and the target is below 0.05.
- The size of $s$ is selected such that at least *40* neighbors can be provided to evaluate *dq*.
- *W* is such that $w_{ii} = 1$ if *i* identifies a state variable in *q* , $w_{ii} = .005$ if *i* identifies a 3D coordinate of the arm end-point, $w_{ij} = 0$ otherwise.

The results of this test are reported in Figures 3, 4 and 5. Clearly, for *60000* learning samples, the map *f* is apparently well modeled, since the residual error rate is low (about 5%) and very few improvements are gained when increasing *N*. Furthermore, the average time to target is more or less asymptotically reached for *N=80000*, while the average residual distance to target when an error occurs increases slowly.

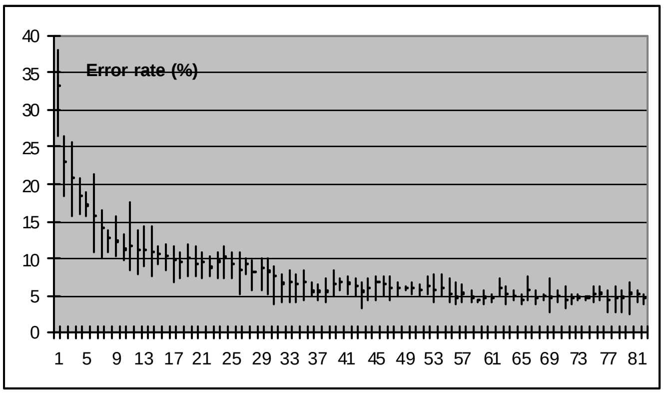


*Fig. 3: The maximum, minimum, and average error values evaluated on the 5 learning runs are presented for N varying from 1000 samples to 85000 samples.*

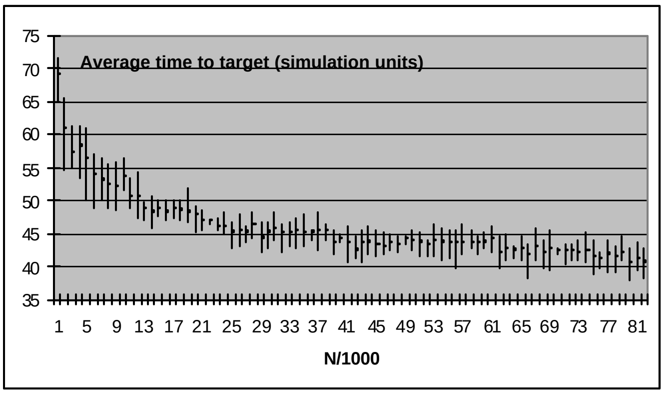


*Fig. 4: The time average to target as a function of N. The maximum, minimum and average values of the 5 learning runs are shown.*

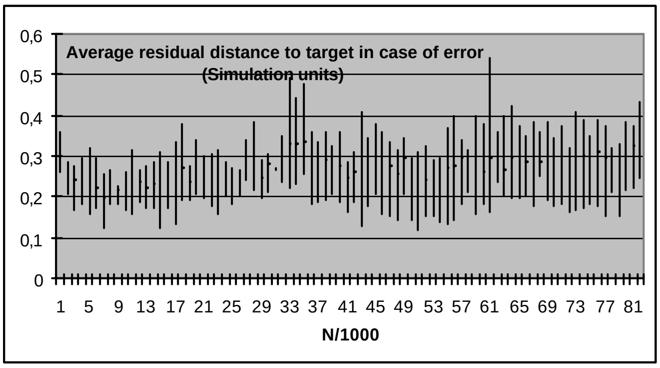


*Fig. 5: The average of the residual distance to target is presented as a function of N. The maximum, minimum and average values of the 5 learning runs are shown.*

The first interesting result is that *N* can be chosen quite low for acceptable performances. It is commonly accepted that for estimating a multivariate function with *10* variables (e.g. *7* degrees of freedom and 3D coordinates) using a kernel density estimator requires above *800,000* samples adequately selected [11]. *N=80,000* seems to be sufficient for the considered task. One reasonable explanation is that the sensory motor loop performs a time average over successive gradient values that compensate small errors due to the coarse estimation. A rough gradient mapping estimation is consequently for the reaching task that is addressed.

To evaluate how well the learned map performs, the *5%* residual error rate should be compared to the performance of a sensory motor loop that incorporates a true gradient operator, namely a gradient value that is analytically calculated. Surprisingly, the error rate for this configuration is about *12.5%*. Note that this figure is met by ASMCM for *N @ 9000* learning samples, which is quite small. This figure is significantly worse than the one exhibited by ASMCM. We cannot propose yet a definitive explanation for this result. Nevertheless, one can conjecture that the noise induced by the coarse gradient map involved in ASMCM allows to escape from local minimums of the error function *E*. If this can be proved, one can view ASMCM as an efficient alternative method to minimize multivariate non linear quadratic functions.

The following figures (Fig. 6 and Fig. 7) show that the learned map with *N=85000* samples exhibits quite similar 3D cinematic outputs to those produced by a sensory motor loop that integrates a true gradient operator (SMCM). Three targets and initial conditions have been selected. Figure 6 shows the evolution of *E* as a function of time (simulation units) for the ASMCM (ErrL) and for the SMCM (ErrG) .

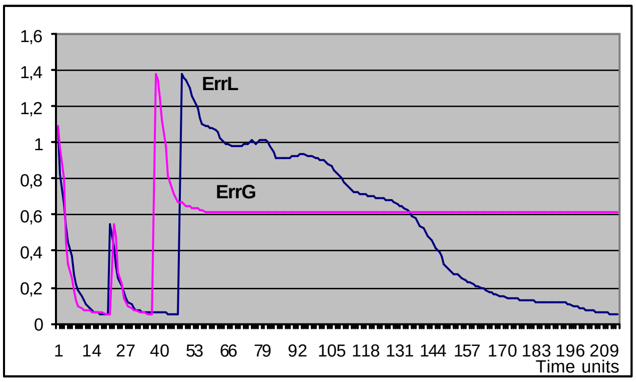


*Fig. 6: Error function E for ASMCM (ErrL) and for SMCM (ErrG). The third target is reached by ASMCM while SMCM falls into a local minimum.*

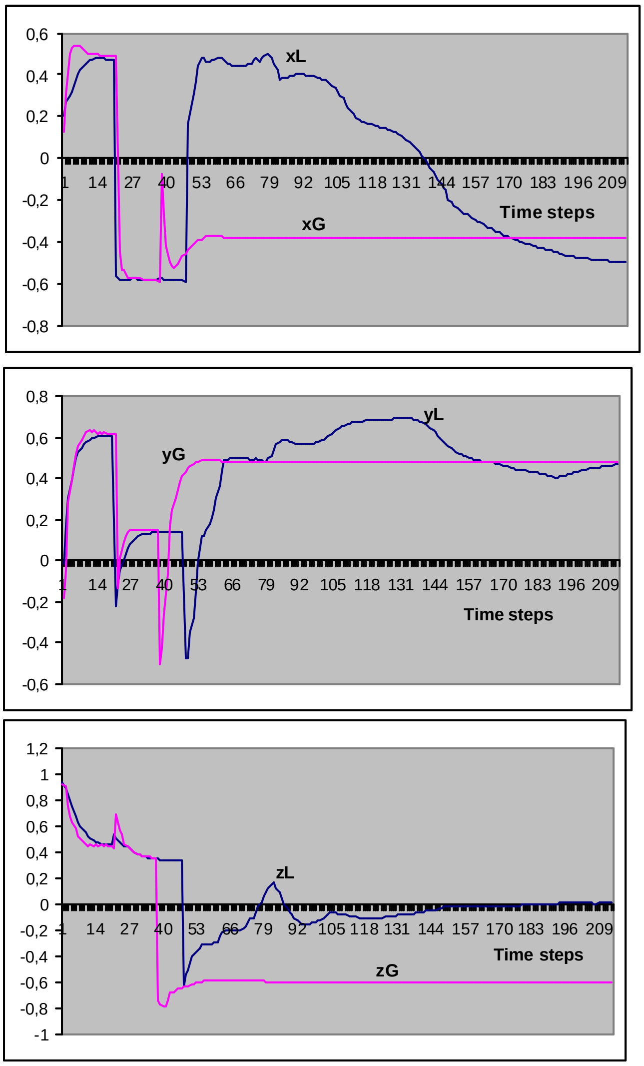


*Fig. 7 : 3D positions of the arm extremity for ASMCM (xL, yL, zL) and SMCM (xG, yG, zG) as functions of time for the three targets reaching tasks.*

The third target cannot be reached by the SMCM that falls into a local minimum, while the ASMCM manage to find a path to this target, escaping local minimums. This is shown on the ErrL signal that is not monotonically decreasing and can be locally increased for ASMCM, while ErrG signal is necessarily monotonically decreasing for SMCM. Figure 7 shows that the 3D *(x, y, z)* position for the two first targets evolve quite similarly for ASMCM and SMCM.

Finally, to exemplify the flexibility of the learned map *f*, we present two last experiments. In the first experiment (Fig. 8 (i)) three spatial targets *a1[0.5, 0.5, 0], a2[0, 0.5, 0.5]* and *a3[0.8, 0, 0.8]* are successively activated. The two first targets are asymptotically reached, while the third one, out of hand, cannot be reached by the arm. In the second experiment (Fig. 8 (ii)), the same targets $a_1, a_2, a_3$ are activated, but the last joint length of the arm is doubled to simulate the adjunction of a stick. Without any update of the learning, all three targets are now reached. This nice property results directly from the nature of the map *f*, that capture directional modification vectors *dq* rather than absolute state vectors *q*.

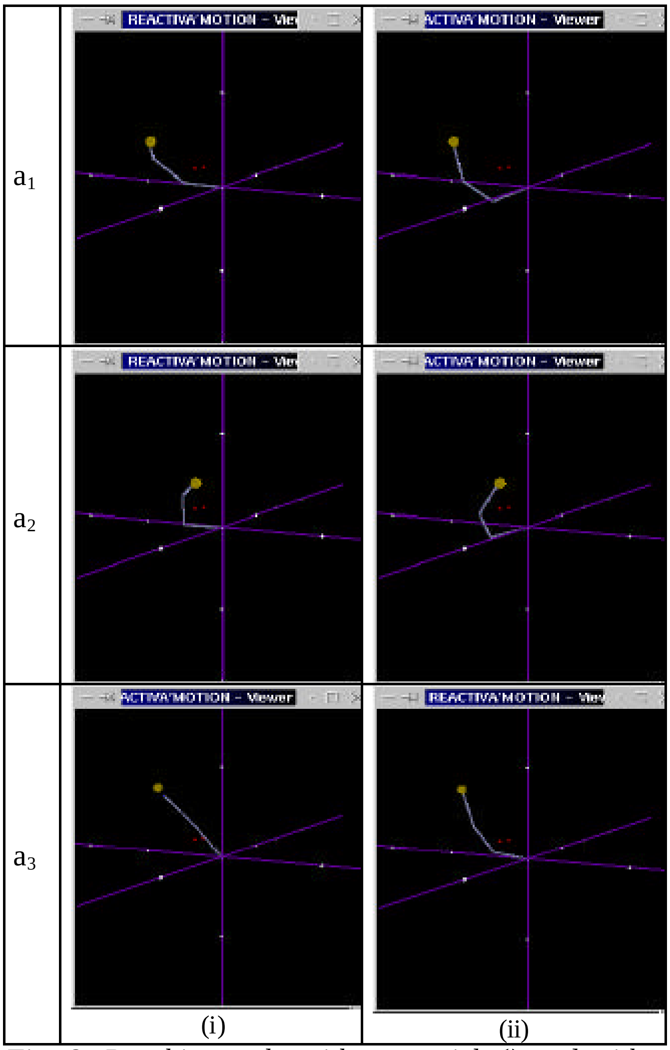


*Fig. 8: Reaching tasks without a stick (i) and with a stick (ii) that double the length of the last joint of the articulated chain.*

# 6 Conclusion

A Sensory-motor control model (SMCM), combining non linear sensory motor transforms is tackled to solve in an iterative fashion the inverse cinematic problem according to a gradient descent strategy.

From this model, the learning of sensory to motor maps involved in sensory motor controlled systems has been addressed at the light of non parametric learning approaches, based on a variable kernel density estimator. Despite the apparent high memory requirement needed by this kind of estimator, the proposed learning scheme efficiently performs when used to control multi-articulated chains with 7 degrees of freedom. This result is obtained even if the number of learning samples is significantly below commonly accepted statistical figures. Nevertheless, the number of degrees of freedom cannot be too high in order to cope with the memory needs of the model.

Since the knowledge of the analytical equations that drive the mechanical or geometrical system are not required, the above described ASMCM model is generic and could be easily adapted to various kinds of sensory motor systems. Neuro-physiology arguments could be found in the literature that support ASMCM like models. Some insights can certainly be gained by confronting ASMCM characteristics with some results obtained in neuro-sciences. This method might help to understand further the sensory motor apparatus and could find applications in robotics and animation of virtual characters.

Finally, ASMCM seems to show better performances than the sensory motor control model which integrates the true gradient operator. The residual learning noise in ASMCM allows effectively to escape local minimums of the error signal to minimize. This property potentially positions the ASMCM as an efficient alternative method to minimize multivariate quadratic functions, according to a pseudo steepest descent strategy.

*References:*

[1] Kawato M., Maeda Y., Uno Y., Suzuki R.. Trajectory Formation of Arm Movement by Cascade Neural Network Model Based on Minimum Torque Criterion. *Biological Cybernetics*, vol. 62, 1990, pp. 275-288.

[2] Bullock D., Grossberg S., Guenther F.H. A Self-Organizing Neural Model of Motor Equivalent Reaching and Tool Use by a Multijoint Arm. *Journal of Cognitive Neuroscience*, vol. 54, 1993, pp. 408-435.

[3] Jordan M.I. Computational motor control. In M. S. Gazzaniga (Eds.), *The cognitive neurosciences. Cambridge*, MA: MIT Press, 1995 pp. 587-609.

[4] Liegeois A. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 7(12) 1977. pp. 868-871.

[5] Tevatia G., Schaal S. Inverse Kinematics for Humanoïd Robots. *IEEE International Conference on Robotics and Automation* (ICRA 2000), 2000.

[6] Gibet S. and Marteau P.F., A Self-Organized Model for the Control, Planning and Learning of Nonlinear Multi-Dimensional Systems Using a Sensory Feedback, *Journal of Applied Intelligence*, Vol.4, 1994, pp. 337-349.

[7] Duda, R. O., & Hart, P. E *Pattern classification and scene analysis*. New York, NY: Wiley, 1973.

[8] Schaal, S. (in press). "Nonparametric regression for learning nonlinear transformations." In: Ritter, H., Holland, O., Möhl, B. (eds*.). Prerational Intelligence in Strategies*, High-Level Processes and Collective Behavior. Kluwer Academic

[9] Churchland, P. S. and SeSejnowski, T. J, *The computational brain*, MA:MIT Press, 1992.

[10] Friedman, J.H., Bentley J.L. and Finkel R.A., "an algorithm for finding best matches in logarithmic expected time", *ACM Trans. Math. Software*, 3, 209-226, 1977.

[11] Silverman B.W., *Density Estimation for Statistics and Data Analysis*, Monographs on Statistics and applied Probability 26, third edition, Chapman and Hall Ed., 1992